MIKROSZÁMÍTÓGÉP MAGAZIN

NEUMANN JÁNOS SZÁMÍTÓGÉP-TUDOMÁNYI TÁRSASÁG LAPJA

1984/4

IZOT 0220 M2 mikroszámítógép
Kulcsra kész rendszerek.
A berszerzéstől a betanításig.

OKISZ SZSZV

MIKROSZÁMÍTÓGÉP MAGAZIN

A Neumann János Számítógéptudományi Társaság lapja

A kiadvány a Tudományszervezési és Informatikai Intézettel együttműködve készül

A szerkesztő bizottság vezetője:
Kovács Győző

Munkatársak:

Broczkó Péter (hírek)
Buday György István (személyi számítógépek)
Garádi János (agyafúrmány)
Jakab Ágnes (ember–gép kapcsolat)
Kovács Győző (az olvasó írja)
Lindner László (sakkprogramozás)
Nacsa Sándor (termékismertető)
Pataki Ernő (programozástechnika)
Petróczy Judit (új könyvek)
Pogány Csaba (alkalmazástechnika, tanfolyam)
Sieben Nándor (százlábú)
Simonyi Endre (klub)
Takácsy Ildikó (favágás)
Varga András (iskola – számítógép)
Vass Nándor (alkalmazások)
Votisky Zsuzsa (játékprogramok)

Felelős szerkesztő:
Könyves Tóth Pál

Szerkesztőség:
Budapest I., Fő u. 68.
Telefon: 154-250

Kiadja: a Lapkiadó Vállalat
Felelős kiadó:
Siklósi Norbert vezérigazgató
Kiadóhivatal:
Budapest VII., Lenin krt. 9–11.
Postacím:
1906 Budapest, pf. 223.
Telefon: 429-350, 221-285

Terjeszti: a Magyar Posta.
Előfizethető
bármely postahivatalban,
a kézbesítőknél,
a Posta hírlapüzleteiben
és a Posta
Központi Hírlap Irodában
(Budapest V.,
József nádor tér 1.,
postacím: 1900 Budapest)
közvetlenül
vagy postautalványon,
valamint átutalással
a PKHI 215–96162
pénzforgalmi jelzőszámra.
Előfizetési díj:
egy évre 168,– Ft,
fél évre 84,– Ft.

Szedte:
a Nyomdaipari Fényszedő Üzem
(847598/09)

Nyomás:
Petőfi Nyomda, Kecskemét,
Külső Szegedi út 6.
(84.41780)
Telefon: 20466
Felelős vezető:
Ablaka István igazgató

ISSN
0236–6088

Címképünk:
A TZ 80 professzionális számítógép.
Műszertechnika GM

Tartalom

Nyitott szerkesztőség	2
(Nem) szent, (de néha) teher	2
Nicolas Schöffer	11
Clive lovag és az óriás	14
Adok – veszek – cserélek	21
Mikroszámítógépek és rádióamatőrök	40
SZEMÉLYI SZÁMÍTÓGÉPEK	
Mi a személyi számítógép?	3
ISKOLA – SZÁMÍTÓGÉP	
Toldi Miklós és a személyi számítógép	6
Számítástechnikai verseny Szolnokon	7
TANFOLYAM	
Alapozás V.	9
PROGRAMOZÁSTECHNIKA	
Ismét a FORTH-ról	12
ALKALMAZÁSTECHNIKA	
Módszerek, fogások, tevékenységek, példák	18
A tártakarékosságról	18
TERMÉKISMERTETŐ	
A TZ 80 személyi számítógép	19
EMBER – GÉP KAPCSOLAT	
Kornyezetunk vedelmerol	22
Pont, pont, vesszőcske...	22
Tudja a magyar ábécét!	23
SZÁZLÁBÚ	24
µKLUB	
Ismeretlen kódok	28
Építsünk számítógépet! III.	29
Mi már építünk!	31
Egy BASIC editor hiba	31
JÁTÉKPROGRAMOK	
A második díjas játékprogram	33
A harmadik díjas játékprogram	34
AZ OLVASÓ ÍRJA	36
FAVÁGÁS	39
SAKKPROGRAMOZÁS	
Lépésről lépésre	43
Sakkozó számítógépek	44
KÖNYVEK	45
HÍREK, ÉRDEKESSÉGEK	47

Nyitott szerkesztőség

„Amikor szembefordulsz
a sodródó jelekkel,
egy percre megbocsátod,
hogy nem rólad beszélnek,
hagyod a névtelent,
a megszámlálhatatlant,
hadd töltse be esendő
törvényeit a térnek."
(Fodor András)

Olvasva a leveleket, mert szerencsére postánk egyre tekintélyesebb, úgy látszik, sikerült szoros és talán őszinte kapcsolatot kialakítani olvasóinkkal. Leveleink nagyobbik részében dicsérő és elmaraszta lő kritikákat kapunk – az utóbbiakat rövidítve az „Az olvasó írja" rovatunkban folyamatosan közöljük. A levelek másik részében olvasóink tanácsokat, javaslatokat adnak, amelyekből nem egyet már meg is valósítottunk. Néhány levél írója – nekik most szeretnék válaszolni – a szerkesztőség munkájára kíváncsi; terveinkről, jövőbeni elképzeléseinkről kérdeznek, szeretnék tudni, hogyan is készül a µM.

Azt hiszem, hogy erre az utóbbira a legkönnyebb a válasz: nehezen, mint a lapok általában. A szerkesztőség tagjainak többsége – velem együtt – a lapszerkesztésben amatőr, csak három munkatársunk profi: a felelős, az olvasó- és a tördelőszerkesztő.

A csapat csupa önkéntesből áll, akiket tulajdonképpen hívni sem kellett. Amikor meghallották, hogy lesz µM, jöttek maguktól, mert szeretik a szakmát és hittek lapunkban.

Egyik olvasónk írja: úgy látszik, mintha a szerkesztőség határozatlan volna, mintha nem tudná, hogy kinek írja a lapot, úgy érzékelte, mintha tartalmi gondjaink is lennének. Van ebben valami igazság. A lap alapításakor úgy határoztunk, hogy írásaink elsősorban az érdeklődő közönségnek szólnak, úgy írunk, hogy azt bárki különösebb számítástechnikai képzettség nélkül is megérthesse, de azért szakmai igénnyel, mert titkon arra is számítunk, hogy lapunkat a számítástechnikusok is szívesen forgatják.

Az eddigi reflexiókból úgy látszik, hogy mindkét célt megközelítettük. Akik vitatják, azok vagy a magasabb tudományos szintet kérik tőlünk számon, vagy pedig – ez a másik véglet – a még oldottabb hangvételt, egy kicsit a bulvár lapok stílusát követelik. Mi – szándékosan – valahol a kettő között jelöltük ki a magunk helyét, mert sem az Információ – Elektronikával, sem a BITLET-tel nem akarunk konkurrálni.

Rovatokat alakítottunk ki, de egyiket sem szeretnénk „vitamin-injekciókkal" mesterségesen fenntartani. Ha nem kapunk, vagy a rovatvezető nem írat elegendő és tartalmas cikket, akkor az egyik vagy a másik rovatot ideiglenesen szüneteltetjük, esetleg meg is szüntetjük. Új rovatot is szívesen indítunk, ha az tartalmilag illeszkedik a µM-nal kapcsolatos elképzeléseinkhez. Olyan rovatot nyilvánvalóan tudatosan nem indítunk, amelyik egy-két megjelenés után kifullad. Az ilyen cikkeket, ha érdekesek és jók, akkor rovaton kívül közöljük.

Cikkeink szerzőit főleg a rovatszerkesztők toborozzák. Viszonylag kevés írást kapunk kérés nélkül. A nálunk tapasztaltabb szerkesztők azt mondják, hogy ez a szokás, ne csodálkozzunk. Én mégis azt szeretném, ha a postás néhány cikket is hozna, nemcsak leveleket.

Örömmel mondhatom, hogy programokat kapunk, elsősorban diákoktól, a Játékprogramok rovat majdnem teljes egészében a beküldött anyagokból él. Egy diák meg is írta, hogy „fogalmazni nem tudok, a helyesírással is hadilábon állok, de talán a beküldött programot használni lehet ...". A program kiváló volt, közöljük is, a programleírást viszont újra kellett írni. Sajnos.

Úgy véljük, hogy a µM – néhány olvasónk bosszúságára – a programok és a számítógépépítés mellett állást kell, hogy foglaljon a számítástechnika és elsősorban a mikroszámítógép-alkalmazás kérdéseiben is; javaslatokat kell tennünk, és bemutatva kritizálni általunk negatívnak tartott jelenségeket. Szeretnénk, ha a lap hasábjain vita alakulna ki a szakma problémáiról, például: hogyan alakul a számítástechnikusok és más területeken dolgozó szakemberek kapcsolata. Ha ezekről a kérdésekről csak a szerkesztők írnak, akkor a lap előbb vagy utóbb érdektelenné, és ami még rosszabb, unalmassá válik. Mi főleg az olvasók véleményét várjuk, és persze, közöljük.

Jól tudjuk, hogy az évi hatszori megjelenés miatt nem tudunk olyan aktuális, friss cikkeket közölni, mint például a hetilapok. Ráadásul egy-egy számot három-hat hónap előretartással szerkesztünk. Én ezt a vezércikket például június elején írom, de ez a lap csak valamikor szeptemberben kerül az olvasóhoz. Ráadásul az 1984/1. szám a hivatalos lapengedélyre várás miatt négy hónapot késett, ami azután további zavarokat okozott (a rejtvényrovat határidői stb.).

A lap fényszedéssel készül, szedés után már nagyon nehezen, de inkább sehogy sem tudunk a kész szövegen változtatni; nem bocsánatos bűneinknek ez a – talán elfogadható – magyarázata.

Úgy tervezzük, hogy az év eleji késés ellenére ebben az évben mind a hat szám megjelenik. A múltkori 15 ezres példányszám után az ez évi első szám 25 ezer példánya – a Posta szerint – három nap alatt elkelt. Ha lenne helyem, egyszer megírnám az újságárusok véleményét, akiket a megjelenés után kérdezgettem. Ti. elmondtam, hogy valami közöm van a laphoz, ezért aztán a kevés példány miatt, hogy úgy mondjam, egy kicsit megkritizáltak.

A második számot már 35 ezer példányban rendeljük, ennyi ugyanis a jelenlegi felső korlát. Remélem, jövőre a példányszámemelést, és ha a szerkesztőség főállású munkatársai is megkezdik a munkát, akkor a havi megjelenést is megkérjük. Szeretném, ha javaslatunkat az olvasók véleményével is alátámaszthatnánk, mert akkor biztosan megkapjuk az engedélyt.

Végezetül – mert a vezércikk terjedelme is korlátozott – úgy tervezzük, hogy az olvasóink által beküldött, de a lapban hely hiányában nem közölt programokat külön kötetben adjuk ki. Ez a könyvecske lesz a µ Könyvtár első kötete, amelyet remélem több is követ. Talán egy másikban a Kit-számítógép leírását hozzuk, esetleg a számítógép-építésről, hardverötletekről állítunk össze egy összefoglaló gyűjteményt.

Terveinkről egyelőre ennyit tudunk elmondani. Mondanivalómat talán nem árt, ha megismétlem. Nyitott szerkesztőség vagyunk, létünk lényege az együttműködés az olvasókkal. Ezt kéri valamennyiüktől is

KOVÁCS GYŐZŐ

(Nem) szent, (de néha) teher

Örültem a lapjukban megjelent, „szent tehén" címszó alá sorolható írásoknak. Az ezek alapján kialakuló képet szeretném egy kicsit kiegészíteni.

Egy Budapesttől 20 kilométerre fekvő kisvárosban két gimnázium, egy szakközépiskola és egy megyei művelődési központ működik, tehát az ismert nyilatkozatok szerint 1984 elején minimum négy darab HT–1080Z típusú gépnek kellene lennie a városban. Ehelyett azonban csak kettőt fedezhet fel az éber szemlélődő.

1. Egyiküket kb. úgy használják, mint az Apáczai gimnáziumban, csak teljesen elszigetelten – lévén az iskola egyházi.

2. Az állami gimnázium sajnos ABC–80 típusú gépet kapott; ez sokat gyengélkedik, és helyhiány miatt kb. 150 méternyire települt az iskola központi épületétől, így az ügyelet nehézkes. A gép, amikor éppen működőképes, kb. heti 10 órában bekapcsolt állapotban leledzik.

3. A másik HT a szakmunkásképzőben van. Erről a gépről rövid történeti áttekintést nyújtok.

1983. december. A TIT-en keresztül az iskola felkért egy előadásra Mire jók a számítógépek címmel. Gépük ekkor nem volt: helyhiány miatt egy másik iskolában „tárolták". A előadást saját gépemen tartottam meg, 20–25 főnyi hallgatóság – kizárólag diákok – előtt. Javasoltam, hogy ha lesz számítógépük, ismét keressenek meg.

1984. február–március. HT–1080Z érkezik – becsomagolva.

1984. április. Egy általános iskola (ahol a gimnáziummal kb. azonos tevékenység folyik – kölcsöngépekkel) többször próbálja kölcsönkérni a HT-t, pontosabban kezdeményez egy gépidőprogram-cserét. Nem jött össze.

1984. május. Kegyes fondorlattal sikerült megszerezni egy napra a gépet. Számlálóállás 000, a billentyűkön enyhe zsírosodás sincs.

4. A művelődési központ még nem kapott gépet, pedig több fórumon is kísérletezett. Évek óta létezik egy logikai játékszakkör: gyermeknapon, juniálisokon 20–150 főnek szerveznek bemutatókat.

A fenti arányt – negyedrész kiváló, negyedrész lelkesedés morzsolta bürokrácia, negyedrész diszkrét semmittevés és negyedrész egyértelmű eszközhiány – lényegében megerősítette a múlt évi Bolyai-kiállítás is. Valójában ez motiválta írásomat is: tipikusnak tartom.

Egyetlen iskolában dolgozó pedagógust sem szándékoztam megbántani, kritizálni. Csupán azt szerettem volna jelezni, hogy 1600 (?) gép szétszórása a feladatnak csak az egyik (kisebbik) fele. A másik felét ki csinálja?

TÖRÖK TURUL

•

Az Iskolaszámítógép rovat vezetőjének megjegyzése:

1. Jelenleg csak nagyon kevés művelődési házban található iskolaszámítógép.

2. Tudomásom szerint minden olyan iskola, amely ABC–80-nal rendelkezik, kapott vagy kap HT gépet. Pest megyében az újabb iskolaszámítógépeket júniusban osztották ki.

3. Örülnénk, ha pontosan tudnánk, melyek azok az iskolák, ahol nem használják kielégítő módon, illetve intenzitással a számítógépeket. A megyei pedagógus-továbbképző intézetek és a Fővárosi Pedagógiai Intézet szakfelügyelői megfelelő támogatást tudnak adni, az Országos Oktatástechnikai Központ tanfolyamokat tarthat, a megyei Neumann-szervezetek közvetlenül segíthetnek stb.

Mi a személyi számítógép?

A személyi számítógép vásárlásánál igen fontos szempont, hogy milyen kész szoftvertermékek kaphatók a géphez. Ezúttal két, a nyugati piacokon slágerterméknek számító alkalmazói programot ismertetünk.

Easy Writer

Az Easy Writer általános szövegszerkesztő program. Segítségével leveleket, dokumentációkat lehet a személyi számítógépen írni, javítani, nyomtatón kiíratni. A megírt anyagok mágneslemezen megőrizhetők, visszahívhatók.

A felhasználót a képernyő tetején megjelenő „menü" tájékoztatja az egy-egy billentyűvel kiváltható funkciókról. A módosítások végrehajtását segíti a tartalom szerinti keresés és átírás lehetősége. Beállítható az egy lapra nyomtatandó sorok száma és a sortávolság. Az ábrák számára szükséges helyet ki lehet hagyni. A margó és az egy sorban elhelyezhető jelek – betük, számok stb. – száma beállítható. Lehetőség van tabulátorok használatára. Írás és beszúrás közben, ha egy szó a sorban már nem fér el, a gép automatikusan az egész szót átviszi a következő sorba, így egy soron belül a felhasználónak nem kell figyelnie, hogy mikor ér a sor végére.

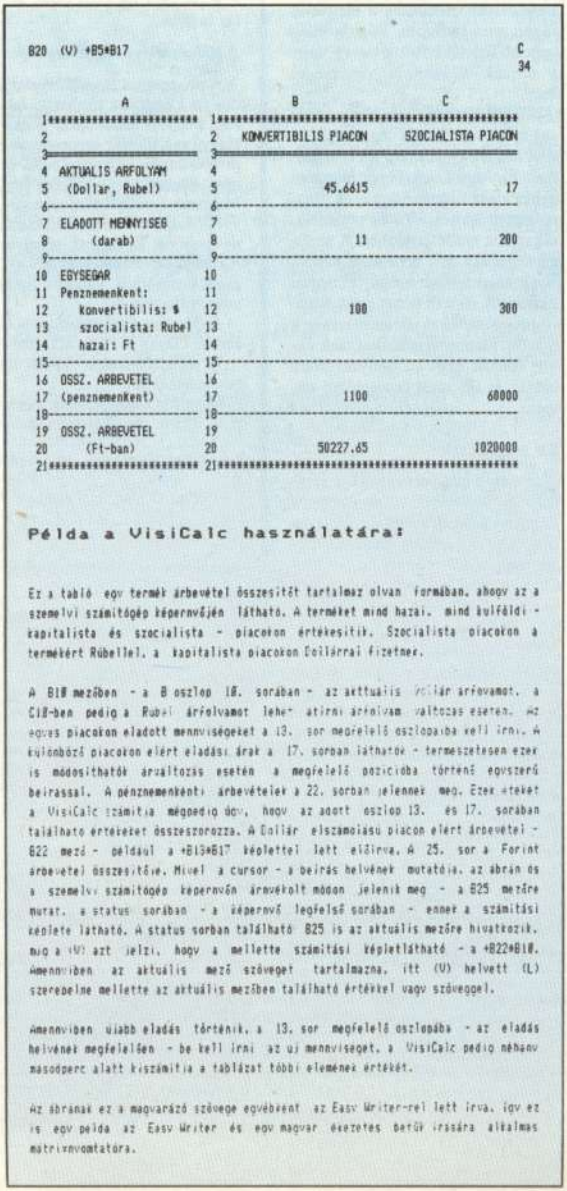
Kérni lehet a megszerkesztett, kijavított anyag kiigazítását, minek hatására a gép a szöveget úgy rendezi – betűközöket elhelyezve a szavak között –, hogy a szövegsorok jobb és bal széle egy vonalba kerüljön. Mód van nagyobb szövegrészek és sorok törlésére vagy beszúrására, máshová helyezésére. Egy levelet akár több hajlékony mágneslemezen elhelyezett levélből is „összevághatunk". Szavakat egyenként vagy a teljes anyagban egyszerre kicserélhetünk vagy törölhetünk.

A szövegben elhelyezett speciális parancsokkal lehet a nyomtatást vezérelni. Beállítható a sortávolság, lapszámozás kérhető, minden lapra azonos fej- és lábrész nyomtatható, nyomtatás közben a lap aljának elérésekor a nyomtatás megállítható. (Ha valaki normál papírra nyomtat, ilyenkor cserélhet vagy fordíthat lapot a nyomtatóban.)

Mátrixnyomtató használata esetén különleges betűformák alkalmazhatók: dupla méretű, dupla vastagságú, összenyomott – fél szélességű – betű, kétszer nyomtatott betű és ezek kombinációja. Ezáltal bizonyos szövegrészeket, címeket stb. ki lehet emelni, kevésbé fontos szövegeket apró betűvel lehet megjelentetni. Mód van tetszőleges szövegrész aláhúzására, félsoremelésre, ami jól használható például matematikai képletekben, alsó és felső indexek nyomtatására.

Az elkészített anyagok mágneslemezen megőrizhetők. Ezzel lehetővé válik például tipuslevelek készítése. Az elkészített levelek minimális munkával – például nevek, címek cseréjével – ismét felhasználhatók, így mindenkinek személyre szóló levelet lehet küldeni.

Mindez azt mutatja, hogy a személyi számítógép e program segítségével egy normál írógépnél sokkal többet, minőségileg újat nyújt. Szinte a nyomdai szedőgép lehetőségeit biztosítja.



B20 (V) +B5*B17

C
34

	A	B	C
1	**********	**********	**********
2		KONVERTIBILIS PIACON	SZOCIALISTA PIACON
3	==========	==========	==========
4	AKTUALIS ARFOLYAM		
5	(Dollar, Rubel)	45.6615	17
6	----------	----------	----------
7	ELADOTT MENNYISEG		
8	(darab)	11	200
9	----------	----------	----------
10	EGYSEGAR		
11	Penznemenkent:		
12	konvertibilis: $	100	300
13	szocialista: Rubel		
14	hazai: Ft		
15	----------	----------	----------
16	OSSZ. ARBEVETEL		
17	(penznemenkent)	1100	60000
18	----------	----------	----------
19	OSSZ. ARBEVETEL		
20	(Ft-ban)	50227.65	1020000
21	**********	**********	**********

Példa a VisiCalc használatára:

Ez a tábló egy termék árbevétel összesítő tartalmaz olyan formában, ahogy az a személyi számítógép képernyőjén látható. A terméket mind hazai, mind külföldi - kapitalista és szocialista - piacokon értékesítik. Szocialista piacokon a termékért Rubellel, a kapitalista piacokon Dollárral fizetnek.

A B18 mezőben - a B oszlop 18. sorában - az aktuális dollár árfolyamot, a C18-ban pedig a Rubel árfolyamot lehet átírni árfolyam változás esetén. Az egyes piacokon eladott mennyiségeket a 13. sor megfelelő oszlopaiba kell írni. A különböző piacokon elért eladási árak a 17. sorban láthatók - természetesen ezek is módosíthatók árváltozás esetén a megfelelő pozícióba történő egyszerű beírással. A pénznemenkénti árbevételek a 22. sorban jelennek meg. Ezek értékét a VisiCalc számítja mégpedig úgy, hogy az adott oszlop 13. és 17. sorában található értékeket összeszorozza. A Dollár elszámolású piacon elért árbevétel - B22 mező - például a +B13*B17 képlettel lett előállítva. A 25. sor a Forint árbevétel összesítője. Mivel a cursor - a beírás helyének mutatója, az ábrán ós a személyi számítógép képernyőn árnyékolt módon jelenik meg - a B25 mezőre mutat, a status sorában - a képernyő legfelső sorában - ennek a számítási képlete látható. A status sorban található B25 is az aktuális mezőre hivatkozik, míg a (V) azt jelzi, hogy a mellette számítási képletlátható - a +B22+B18. Amennyiben az aktuális mező szöveget tartalmazna, itt (V) helyett (L) szerepelne mellette az aktuális mezőben található értékkel vagy szöveggel.

Amennyiben újabb eladás történik, a 13. sor megfelelő oszlopába az eladás helyének megfelelően - be kell írni az uj mennyiséget, a VisiCalc pedig néhány másodperc alatt kiszámítja a táblázat többi elemének értékét.

Az ábrának ez a magyarázó szövege egyébként az Easy Writer-rel lett irva, igy ez is egy példa az Easy Writer és egy magyar ékezetes betűt irására alkalmas mátrixnyomtatóra.

VisiCalc

A másik népszerű szoftvertermék a VisiCalc. Ez olyan eszköz, melynek segítségével gyorsan és egyszerűen lehet bonyolult táblókat – például vállalati mérlegtablót – létrehozni és kezelni.

A program egy 255 sorból és 63 oszlopból álló táblázat kezelésére alkalmas. Az oszlopok azonosítása betűkkel, a soroké számokkal történik. A táblázat minden eleméhez – vagyis a sorok és oszlopok metszéspontjaihoz – a gép memóriájában egy ún. rekord tartozik. A felhasználó minden rekordhoz hozzárendelhet egy nevet, egy adatot vagy egy képletet. A táblázat egyes elemeibe szöveges információk is kerülhetnek.

A személyi számítógép képernyőjén nagyobb tabló nem jeleníthető meg egyszerre. A képernyőn, mint egy ablakon keresztül lehet bepillantani a táblázatba. Ez az „ablak" tetszőlegesen mozgatható a táblázat fölött. A számítógép képernyőjén a táblázat tetszőleges

része, akár nem egymás melletti sorai vagy oszlopai is megjeleníthetők. A képernyőre kerülhet például a tabló A oszlopa, amely adott esetben szöveges információkat tartalmaz, és az E–J oszlopok, amelyekkel éppen dolgozni akarunk.

Tegyük fel, hogy egy könyvelő az E oszlop 5. sorába a vállalat dollár-, az E oszlop 6. sorába pedig a rubel-árbevételét kívánja írni. Azt szeretné, hogy az F oszlop 5. sorában megjelenjen a dollár-árbevétel forint megfelelője, míg az F oszlop 6. sorában a rubel-árbevétel forintértéke. A teljes külföldről származó árbevétel összegét pedig az F oszlop 7. sorából szeretné kiolvasni. Ekkor az F oszlop 5. sorához a +E5*45.4525 – a dollár-árbevétel szorozva a dollár-árfolyammal –, az F oszlop 6. sorához a +E6*17 – a rubel-árbevétel szorozva az aktuális rubel-árfolyammal –, az F oszlop 7. sorához pedig a +F5+F6 képletet kell rendelnie. Ha ezután bármikor az E5-be beír egy dollárösszeget, az F5-ben pillanatok alatt ennek forintra átszámított értéke, az F7-ben pedig az e számított értéknek és az F6-nak az összege fog megjelenni. Ha pedig E6-ot módosítja – a rubelárbevételt –, az F6-on és az F7-en fog új érték kiíródni. Az a könyvelő, aki a VisiCalc-ot használja, biztos lehet, hogy minimális időráfordítással naprakész, garantáltan pontos eredményeket tartalmazó mérleget kap.

Még nyilvánvalóbb e programtermék hasznossága, ha olyan táblót készítünk vele, amely döntéselőkészítésben segít. Tegyük fel, hogy egy vállalatigazgató egy tárgyaláson vesz részt, ahol ajánlatot kell tennie partnerének egy egyedi termék előállítására. Tegyük fel továbbá, hogy a termék előállításának költségét igen sok tényező befolyásolja, amelyek konkrét értékei csak a tárgyaláson derülhetnek ki. Amennyiben a táblót úgy készítettük el, hogy ezeket a paramétereket a megfelelő helyre beírva, a termék előállításának árát a gép képes kiszámítani, vezetőnk biztos lehet abban, hogy cégének a termék előállítása a gép által kalkulált áron nyereséges lesz. Amíg képzeletbeli vezetőnk nem rendelkezett ilyen eszközzel, a tárgyalás végén nem tudhatta, hogy a világ legjobb üzletét kötötte-e meg, vagy a termék ellőállítása veszteséges lesz.

A fenti két program használatát egy általános iskolát végzett ember is el tudja egy-két nap alatt sajátítani. Egy-egy konkrét esetben nem nehéz kiszámítani, hogy ezek használata mekkora idő- és pénzmegtakarítást eredményezhet. A szoftvertermékek írói arra törekednek, hogy hasonló minőségű, könnyen kezelhető, időt és pénzt megtakarító programokkal jelenjenek meg a piacon, hiszen terméküket csak akkor tudják nagy példányszámban eladni, és így saját bevételüket növelni.

És végül...

Ennek a cikksorozatnak a megírásával az volt a célom, hogy a személyi számítógépekkel most ismerkedő olvasónak némi segítséget nyújtsak e még alig tíz éve létező, és máris gazdaságot-társadalmat átformáló eszközről, a személyi számítógépről. Alapvető fogalmakat kívántam tisztázni, a személyi számítógépek főbb elemeit próbáltam vázlatosan ismertetni. Szeretném, ha az olvasó megértené, hogy a személyi számítógép eszköz. Olyan eszköz, amely segít időt és pénzt megtakarítani, képes a munkát megkönnyíteni, minőségét javítani, és partner lehet a szabad idő kellemesebb eltöltésében.

FINTA GYÖRGY

PÁLYÁZATI FELHÍVÁS

Az NJSZT, a szekszárdi Garay János Gimnázium és a Mikroszámítógép Magazin Szerkesztősége pályázatot hirdet programok írására az alábbi kategóriákban:

1. Új, önálló játékprogram készítése (HT–1080Z, ABC–80, PRIMO, AIRCOMP, ZX81, COMMODORE 64 gépre).
2. Valamely tantárgyhoz kapcsolódó, a tanítási órát segítő (szemléltető, számításos, ...), illetve a tanulók önálló tanulását elősegítő, oktatóprogram készítése (a fent említett gépekhez, de itt előnyben részesülnek a HT–1080Z és a PRIMO gépre írt programok).

A pályázaton részt vehet minden általános iskolai és középiskolás tanuló.

A pályázat beadási határideje: 1985. január 15.

A programot BASIC nyelven kell készíteni, mágnesszalag kazettán (a kazettára többször felvéve), rövid leírás kíséretében (mit tud, hogyan működik a játék, illetve hányadik osztály, mely tantárgyának melyik anyagrészéhez kapcsolódik stb., egyéb adatok a programmal kapcsolatban), jeligével ellátva (zárt borítékban a név, lakcím és az iskola) kérjük beküldeni a Mikroszámítógép Magazin Szerkesztőségébe (Budapest I., Fő utca 68. 1027).

A Magazin Szerkesztőségének joga van a pályázaton részt vett programok közlésére, amelyért a szokásos honoráriumot fizeti. A döntő 10–10 résztvevőjét a Társaság tagjaiból, a Garay Gimnázium tanáraiból, illetve a Szerkesztőség munkatársaiból álló előzsüri választja ki.

A döntőre Szekszárdon, a Garay János Gimnáziumban rendezendő Garay Napok alkalmából, 1985 márciusában kerül sor.

A döntőbe jutott 10–10 tanulót a Garay János Gimnázium vendégül látja.

Mindkét kategóriában az első három helyezett programot díjazzuk, illetve mindkét kategóriában kiadjuk a közönség díját.

DR. VÁMOS TIBOR *elnök NJSZT*

ZENTAI ANDRÁS *igazgató Garay János Ált. G.*

KOVÁCS GYŐZŐ *a szerk. bizottság vez. Mikroszámítógép Magazin Szerkesztősége*

PROGRAMOZÁSI RENDSZEREK '84

címmel konferenciát rendez november 26–27-ig Szegeden a Neumann János Számítógéptudományi Társaság Programozási Rendszerek Szakosztálya. A konferencia témája a mikrogépes szoftverfejlesztés, elsősorban szoftverkészítők számára. Részvételi díj: 800,– Ft, jelentkezési határidő: október 10. Jelentkezési lapok a Társaság titkárságán (1054 Bp., Báthori u. 16.) kaphatók.

Most már ez is kell...

Szoftver
COMMODORE számítógépekre

Fejleszt

Rendszer- és felhasználói programcsomagokat
a megrendelők egyéni igényeinek megfelelően
– termelésirányítás
– rendelésnyilvántartás – tervezői programcsomag
– hálózatba kapcsolás

Forgalmaz

Szoftvertermékeket, hardvereszközöket
– egyéni alkotóktól
– gazdasági munkaközösségektől
– vállalatoktól, szövetkezetektől

Elad

FORTH, folyószámla, SORT-MERTE, jogszabály-nyilvántartás, játék, főkönyvi kivonat, gépi kódu monitor és Assembler, nemzetközi tizes billentyűzet, takarmányoptimalizálás, V24 interfész, matematikai statisztikai, termelési programnyilvántartás, döntéssegítő, terminál emulátor, statikai számítások, deviza-keresztárfolyam nyilvántartó rendszer, 80 karakteres bővítő, BASIC oktató sorozat, DATASET magnókiváltás

NOVOTRADE

1136 Budapest,
Fürst Sándor u. 24–26.
Telefon: 310-546, 110-687
Telex: 22-5959 NOVTR H

„COMMODORE 64" és „700", a COMMODORE BUSINESS MACHINES INC. védjegye

Toldi Miklós és a személyi s

Humán tárgyak oktatásában is

Vájú vagy vályú, imuja vagy mulya, ereklye vagy erekje, gólya vagy gója?

– Hogy írjam? – bajlódik a diák.

– Hogy tanítsam? – töpreng a tanár.

Mindenkinek sok nehéz percet okozhatnak az ilyen és ehhez hasonló kérdések. A tanuló gyorsan megunja, ha ötvenszer leíratják vele ugyanazt, a tanárnak pedig nincs mindig ideje és türelme valamely hibát századszor is kijavítani.

Természetesen ezeknek a problémáknak a megoldásához nem akarunk és nem tudunk mindenható tanácsokat adni, de szeretnénk egy olyan új módszerről szólni, amely sokat segíthet a magyar nyelv és irodalom egyes fejezeteinek tanításában. A módszer általánosan alkalmazható, mióta minden középiskolában van személyi számítógép.

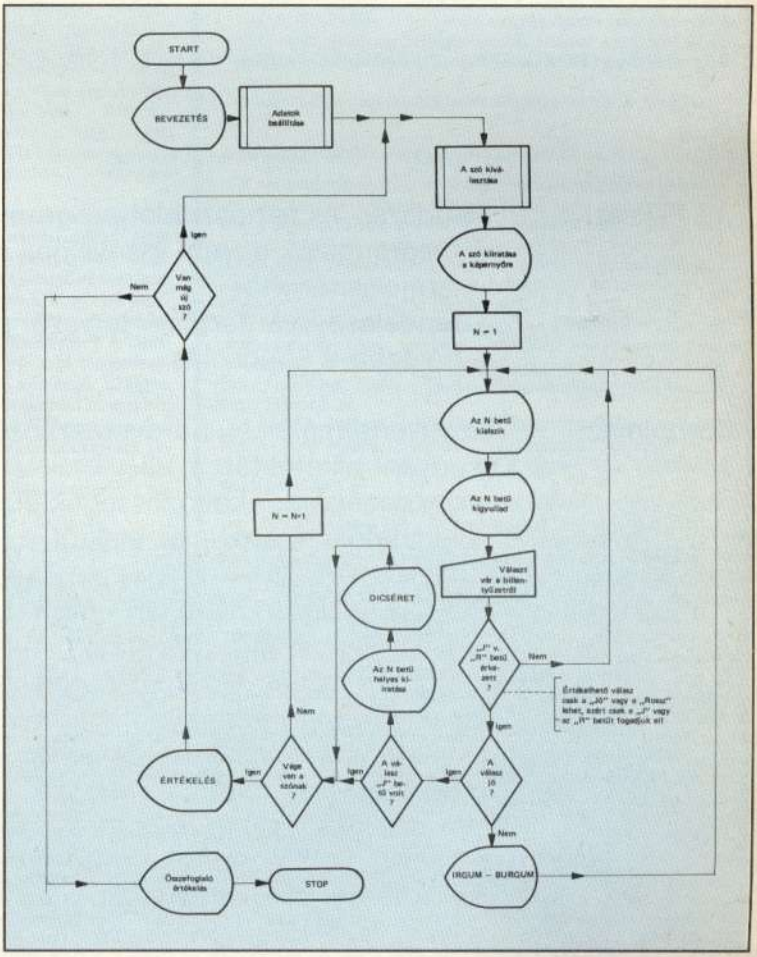
De mit keres a magyarórákon a számítógép? – vetődhet fel sokakban a kérdés. A kétkedőknek válaszul: nem akarjuk forradalmasítani a magyar nyelv és irodalom oktatását, de bemutatunk néhány ötletet ebben a témakörben.

Maradjunk mindjárt a bevezetőben is említett és ahhoz hasonló helyesírási problémáknál: j-vel vagy ly-nal való írás; szóvégi -u, -ú, illetve -ú, -ű; egybeírás – különírás; rövid és hosszú magán-, illetve mássalhangzók; tulajdonnevek tárgyragos alakjai stb.

Ahhoz, hogy a tanulók nagy biztonsággal kikerüljék a helyesírási buktatókat, többségüknek sok gyakorlásra van szüksége, ez pedig akkor lesz igazán eredményes, ha állandó visszajelzést kapnak munkájukról. És ez az a pont, ahol a számítógép jelentős segítséget adhat. Bemutatjuk egy olyan program folyamatábráját, amely szavak helyesírását gyakoroltatja a diákokkal. Ennek megértéséhez elég, ha a személyi számítógépről a következőket tudjuk.

A géppel való kommunikáció a számítógéphez csatlakoztatott, tetszőleges tv-készülék képernyőjén keresztül folyhat. A számítógépnek adott utasításainkat billentyűzeten gépeljük be. Ezek és a számítógép válaszai a tv-képernyőn jelennek meg. A begépelt adatokat és utasításokat a számítógép a memóriájában tárolja, illetve dolgozza fel. Fontos szempont, hogy ezek a személyi számítógépek igen kis méretűek és könnyen kezelhetők.

Az *ábra* segítségével nézzük meg, hogy is működik ez a program! Miután elindítjuk (START), a számítógép képernyőjén egy bevezető szöveg jelenik meg, amely a program használatával kapcsolatos tudnivalókról tájékoztat. Az ismertető után a gép kiírja azt a szót, amelynek helyesírását a tanulónak ellenőriznie kell. A megjelenített szóban lehet egy vagy több helyesírási hiba is.



A képernyőn a szó első betűje villogni kezd. Ez addig tart, amíg a diák el nem dönti, hogy a betű helyesírása „JÓ" vagy „ROSSZ", és ennek megfelelően le nem nyomja a J vagy az R billentyűt. A gép ekkor megvizsgálja, hogy megfelelő volt-e a válasz. Ha nem, akkor megdorgálja a tanulót (IRGUM-BURGUM), és a vizsgált betű továbbra is villog, tehát a diáknak lehetősége van arra, hogy kijavítsa válaszát. Helyes válasz esetén két lehetőség van. Ha a betű helyesírása rossz volt, akkor megjelenik a képernyőn a kijavított változat, és a tanuló dicséretet kap. Ezután, vagy pedig akkor, ha a betű helyesen volt kiírva, elkezdődik a következő betű vizsgálata, az előzőekhez hasonló módon.

Miután a gyerek a szó végére ért, rövid értékelést kap, ami megerősíti az adott szó helyes írását. A program egy újabb szó ugyanilyen típusú vizsgálatával folytatódik. A gyakorlás befejezésekor a számítógép értékeli (például százalékosan) a diák munkáját, ami fontos információ a tanár és a tanuló számára is.

A program egymás utáni többszöri futtatása is érdekes és hasznos egyazon tanuló számára is, mivel a számítógép nagy mennyiségű megvizsgálandó szót tud tárolni, amelyeket szükség esetén a tanár ki is cserélhet. A gép segítségével mód van arra is, hogy a tárolt szókészletből véletlenszerűen válassza ki a program a képernyőn egymás után megjelenő szavakat. Így egy-

számítógép

részt lehetőség van különböző helyesírási problémák gyakorlására, másrészt egy adott témakör alapos és nem monoton megtanulására.

Folyamatábránk elég általános ahhoz, hogy nemcsak a példaként felsorolt, hanem az azokhoz hasonló helyesírási problémák gyakoroltatására is alkalmas programok alapja legyen.

Kis továbbfejlesztéssel elérhető, hogy egész mondatok, illetve szövegrészek helyesírását is gyakoroltassuk. (Az ékezetek kitétele csak programozástechnikai kérdés. Már elkészült ugyan ennek a gépnek olyan változata, amely használ ékezetes betűket, de a rövid és hosszú magánhangzók megkülönböztetését mindenképpen programból kell megoldani.)

A számítógép segítségével játszhatunk is a szavakkal. Erre említünk néhány ideillő példát. Közismert játék és amellett jó nyelvgyakorlás az anagrammakészítés. Írhatunk olyan programot, amely egy adott szó betűit kívánságunknak megfelelően – vagy véletlenszerűen, vagy megadott rendszer szerint – különböző sorrendbe rakja. Itt a gép előnye „csak" az, hogy az átrendezést gyorsan és pontosan hajtja végre, és így nagyobb esélyünk van arra, hogy az átrendezett betűkből ismét értelmes szót kapjunk. Ilyen típusú az a program is, amely a magyar nyelv helyes szórendjét gyakoroltatja a tanulókkal (adott szavakból mondatkészítés, egy mondat szavainak átrendezése stb.).

Az ún. kakukktojás-játékok is könnyen megvalósíthatók a számítógépen. Lássunk erre néhány példát: igék közé kevert főnév; magas hangrendű szavak közé kevert mély hangrendű szó; belső keletkezésű, jövevény- és idegen szavak keverése; ismert verssor egy szavának kicserélése; valamely költő életrajzában idegen adat; valamely író művei között idegen mű stb.

Ezeket a játékokat egyébként is nagy élvezettel játsszák a gyerekek, de a számítógép itt is új színt jelent, hiszen a feladatokat egy gép adja, amely ellenőrzi is a tanuló válaszait.

A számítógép hasznos és hatékony segítőtárs a verstanulásban is. Ennek egyik formája, amikor soronként memorizáljuk a verset. A számítógéptől kikérhetjük a tanulandó verssort (vagy akár az egész versszakot), illetve lehetőség van arra is, hogy a már megtanultat begépelve, ellenőrizzük annak szöveghűségét és helyesírását. Így állandó és közvetlen visszajelzést kapunk arról, hogy mennyire pontosan tudjuk a verset.

Mind verstanuláskor, mind elemzéskor érdekes és jól használható lehetőség a vers ritmikájának számítógépes „megzenésítése". Készíthetünk programot, amely eldönti adott szótagok hosszúságát vagy rövidségét, felismeri az ezekből felépülő verslábakat, majd ezeknek megfelelő hosszúságú és magasságú hangokból összeállítja a vers „zenéjét". A versek elemzését és jobb értését segíthetik olyan számítógépes strukturális vizsgálatok, amelyek egy adott vers magas és mély hangzóinak arányára, az előforduló különböző szófajok gyakoriságára, a rímképletre vonatkoznak. Ezekből a képernyőre áttekinthető összesítéseket, grafikonokat rajzoltathatunk.

A II. gimnáziumi magyar nyelv tankönyv (Honti Mária–Jobbágyné András Katalin: Magyar nyelv II. osztály) végén táblázatokat találunk. Ezek rendszerezik a mondatfajtákat, az igenemeket, a határozókat, a névmásokat, az igetöveket, a névszótöveket stb. Tanulmányozásuk során kimeríthetetlen tárházára bukkanhatunk számítógépes programok készítésének.

Nézzük példaképpen a 182. oldalon lévő, Az ige alakrendszere című táblázatot. Azonnal felvetődik egy olyan program ötlete, amely az igeragozást gyakoroltatja a tanulókkal. Ennek lehet olyan változata is, amelynél adott mondatkörnyezetben lévő igéről el kell dönteni, hogy milyen módban és időben van, valamint hogy milyen ragozású.

A táblázat alapján könnyen készíthető „kakukktojás" is (például várok, várasz, vár, várjuk, vártok, várnak), de helyesírást gyakoroltató program is (a fon, süllyed, edz stb. igék ragozott alakjaiból).

A legtöbb táblázattal kapcsolatban készíthető olyan program, amely adott szavak közül kiválasztatja a tanulóval az azonos típusba tartozókat.

Gyakorlottabb programozók megpróbálkozhatnak olyan konstrukciós programok készítésével, amelyek a diákkal előre megadott szavakból mondatot szerkesztet. A mondatok egyértelmű szerkesztéséhez a képernyőn megjelenő ábrák, rajzok nyújthatnak segítséget.

A gimnáziumi irodalom tankönyvek végén (a IV. osztályos kivételével) a legfontosabb fogalmakat Kislexikonban összegezték. Ezek is – a nyelvtankönyvi táblázatokhoz hasonlóan – sok ötlet forrásai lehetnek.

Befejezésül hagytuk, de talán az egyik legérdekesebb problémának tartjuk Péchy Blanka és Deme László Beszélni nehéz című rádióműsorához kapcsolódó ötletünket. Ebben a műsorban a kitűzött mondatokat kell helyes kiejtésjelöléssel beküldeni a hallgatóknak. Az itt használatos jelrendszer átültethető a számítógép karakterkészletére, és így ilyen feladatok a számítógép segítségével is végezhetők. Ezenkívül kirajzoltatható a mondat hanglejtése, és a jobb megértést segítheti elő, ha a mondat zeneiségét (ritmusát, hangsúlyait) hanggenerátorral a kiejtésjelölések alapján meg is szólaltatjuk.

Bízunk benne, hogy egyre több magyar nyelvi és irodalmi órán rajzol, zenél, tanít, bemutat, ellenőriz, gyakoroltat a számítógép, és ehhez az itt felsorolt ötletek is segítséget nyújtanak.

MAJOR ZOLTÁN
ELTE Matematikai Intézet
VALOVICS ISTVÁN
ELTE Apáczai Csere János Gyakorló Gimnázium

SZÁMÍTÁS-TECHNIKAI VERSENY SZOLNOKON

A Szolnok megyei Pedagógiai Intézet 1984. június 7-én rendezte meg a megye középiskolai tanulói számítástechnikai versenyét, amelyen az iskolai válogatók első és második helyezettjei vehettek részt, összesen 44 tanuló. A verseny két fordulóból állt; a feladatokat Kőhegyi János állította össze.

A verseny első felében a tanulók két óra alatt négy elméleti feladatot oldottak meg írásban. Ezek között volt programkészítés, hibás program javítása, programkészlet hatékonyabbra való átírása. A döntőbe a legjobb 15 versenyző került, akiknek HT–1080Z iskolaszámítógépen három feladatot kellett megoldaniuk.

A verseny végeredménye:

1. Nagy Sándor, II. oszt., Kunszentmárton, József Attila G.
2. Bakki Lóránt, III. oszt., Jászberény, Lehel vezér G.
3. Száz István, III. oszt., Törökszentmiklós, Bercsényi Miklós G.
4. Csatlós Béla, II. oszt., Mezőtúr, Teleki Blanka G.
5. Pócz Zoltán, III. oszt., Jászberény, Erősáramú Szakközépisk.
6. Török Péter, II. oszt., Szolnok, Verseghy Ferenc G.

A zsüri – melynek elnöke dr. Varga András volt – szerint a döntőbe jutott 15 tanuló magas színvonalú tudásról tett bizonyságot.

A verseny megerősítette a megyei Pedagógiai Intézetet abban a véleményében, hogy az elkezdett munkának (pedagógusok 80 órás tanfolyama, szakköri segédanyagok stb.) megvan az eredménye.

HORVÁTH TIBOR
igazgatóhelyettes

Kabalapályázatunkra érkezett

4. *Derda Sándor, Dunakeszi*

floppymat

A VILATI adatelőkészítő, adatfeldolgozó berendezés családjának legintelligensebb tagját a széles periféria-választék, hatékony szoftvertámogatás jellemzi. Előnyös tulajdonsága az ékezetes magyar karakterkészlet. A FLOPPYMAT SP előnyösen alkalmazható az ügyvitel minden területén önálló feldolgozásra, nagyszámítógépes hálózatban, intelligens batch-terminálként.

ALAPKIÉPÍTÉS

Hardver	Központi egység – 8 bites mikroprocesszor – 64 kbájt RAM – 2 kbájt ROM
Képernyő	– karakterkészlet: ékezetes magyar – sorok száma: 16 kisbetűs vagy 9 nagybetűs – karakter/sor: 64, illetve 32
Billentyűzet	– kombinált alfanumerikus (ékezetes magyar) – külön numerikus – funkciós
Háttértár	– 2 × 8" hajlékonylemez (típus: MF6400) (felírási mód: IBM softszektoros)
Interfész	– nyomtató interfész DZM 180, Consul 2111 vagy bármilyen Centronics kompatibilis interfésszel nyomtatóhoz – iker hajlékonylemezes egységhez (harmadik, negyedik hajtás) – 16 bites általános célú I/O (TTL szintű)
Szoftver	XDOS hajlékonylemezes operációs rendszer – szerkesztő és dokumentáló – assembler fordító – linker – betöltő és futtató – hiba-nyomkövető – lemezmásoló és inicializáló – virtuális perifériakezelő
PLF8 fordító és futtató	– decimális fixpontos aritmetika – I/O kezelő – fájlkezelő – tömbkezelő

FMN 09 adatelőkészítő programcsomag	IBM 3741 kompatibilis adatrögzítő – képernyő- és lemezformátum programozás – nyomtatási formátum programozás – programozható összegzés – rendezés lemezen numerikus vagy alfabetikus sorrendbe – lemez- vagy fájlmásolás

OPCIÓK

Hardver	– cartridge lemezillesztő (max. 4 lemez) – ECMA szabvány szerinti mágneskazettaillesztő (adathordozó-konvertálásra) – PROM, EPROM beíró – CCITT V24 interfész
Periféria	– nyomtató DZM 180 Consul 2111 IZOT BPA 1 (margaréta) – iker hajlékonylemez-egység: MF6400 – cartridge lemez: CM 5400 – mágneskazetta: PK–1 – karton előtét DZM 180-hoz
Szoftver	Adatátviteli szoftverek – alapmódú aszinkron, transzparens – alapmódú szinkron, transzparens – BSC 2780 szinkron, transzparens
Alkalmazói rendszer generátorok	– UPDT állománykezelő, karbantartó – TABLB állománylekérdező, tablózó

Ügyviteli szempontoknak megfelelően továbbfejlesztett teljes PASCAL nyelv.

ALKALMAZÓI RENDSZEREK

A VILATI szervező, szoftverfejlesztő vállalatokon keresztül biztosít a felhasználók részére alkalmazói rendszereket.

Jelenleg kész, működő alkalmazói rendszerek:
- vevői rendelésállomány-nyilvántartás
- szállítói rendelésállomány-nyilvántartás
- többraktáros anyag-termék készlet nyilvántartó és elszámoló rendszer
- munkaügyi nyilvántartó rendszer
- mezőgazdasági bérügyviteli rendszer
- szállítójármű-nyilvántartó és menetlevél-feldolgozó rendszer
- belkereskedelmi és szövetkezeti bolti elszámoltató rendszer
- belkereskedelmi számlázási rendszer
- államigazgatási nyilvántartó rendszer
- személyzeti nyilvántartó rendszer

Kidolgozás alatt álló rendszerek:
- szövegszerkesztő és dokumentumkezelő rendszer
- műszaki tevékenység (karbantartás, fejlesztés, beruházás, szolgáltatás) nyilvántartó rendszer
- fogyóeszköz-nyilvántartó rendszer
- főkönyv-folyószámla könyvelési rendszer
- számlázási rendszer (külker, ipari)
- állóeszköz-, műszernyilvántartó rendszer
- utókalkulációs rendszer
- általános bérelszámolási rendszer
- államigazgatási rendszerek
- költségvetési szervek ügyviteli rendszere

A felhasználók rendelkezésére álló rendszerfejlesztő szervezetek:
FLOPPYMAT Fejlesztő Gazdasági Társulás
Képviseleti Iroda: Bp. VI., Szinyei Merse u. 1.
Tagvállalatok: **VSZFT** Bp. XIV., Ajtósi Dürer sor 10.
SZÖVORG Bp. I., Attila u. 13.
MŰSZI Bp. XII., Érmelléki u. 13.
KERSZI Bp. XIII., Dózsa György u. 150.
VILATI Bp. I., Krisztina krt. 55.

További rendszerfejlesztő szervezetek:
ÁSZI Bp. XI., Ménesi út 5.
DATORG Bp. V., Dorottya u. 6.
PM SZÜV Bp. V., Tanács krt. 20.
OKISZ SZSZV Bp. IX., Üllői út 47.
Magyar–Mongol Barátság MGTSZ, Mezőtúr

Forgalmazó: **VILATI**

1253 Budapest 13. Pf. 14.
Telex: VILATI Bp. 22–5042

ügyviteli mikroszámítógép

Alapozás V.

Egy berendezés eredményes felhasználásához ismerni kell annak működéstörvényét. Ha a berendezés több részből áll, akkor e részek működéstörvénye mellett igen fontos az egyes alkotórészek működése összehangolásának rendje is. Mivel minden rendszer, így a szgép is (a hardver is, a programok is és együttesük is) operátorcsapat, az operátorcsapat munkájának megszervezése, azaz az egyes részek működésének összehangolása a következő fontos kérdéskör, amivel foglalkozunk.

Az operátorcsapat elemei működésének összehangolása

A szgépben nagyszámú operátor dolgozik egyidejűleg. Ezek üzemmódjainak vezérlése a gép helyes és gazdaságos működése szempontjából döntő kérdés. Ennek oka az, hogy a jelenlegi szgéptervezési gyakorlat definiál működésmódokat és kihasználja az azokkal való manipulációs lehetőségeket a szgép működésének irányítására. Ne felejtsük: a működésmód a működés valahogyan definiált jellemzője. Lehet például egy kapó kézben pillanatnyilag levő érték, de bonyolultabban előállított függvény is.

A kétértékű működésjellemzők esetében ezek jelölésére gyakran használják a 0 és az 1, illetve a „nem igaz" és az „igaz" értéket. (A „nem működik" és „működik" szóhasználat is gyakori, de ezt félrevezető volta miatt kerüljük.) Használjuk még az előbbi célra a „nem aktív" és az „aktív" szavakat is. Ez szerencsésebb szóhasználat, ugyanis kifejezi azt, hogy nem „nemműködésről", csak másféle működésről, adott szempont szerinti inaktív, passzív, mintegy várakozó, nem használt, tartalékos állapotról van szó.

Egyszerűbb, egy feldolgozóegységgel (egy processzorral) dolgozó gépek esetében programjaink operátorai – programjaink által definiált operátorok – csak egymás után lehetnek aktívak, csak egymás után lehetnek aktív üzemmódban.

Bonyolultabb, több feldolgozóegységgel (több processzorral) működő szgépeknél azonban ugyanannak a programnak az operátorai közül is egyszerre több dolgozhat aktív üzemmódban, hasonlóan magához a szgéphez. Ilyen esetekben döntő kérdés az egyes operátorok aktív üzemmódban levésének figyelése és ezeknek az állapotoknak (folyamatoknak) az összehangolt irányítása, ugyanis a gép helyes működése enélkül nem valósítható meg.

Ilyenkor egyszerre több operátor valamilyen, például „aktív" üzemmódba juttatása az első feladat. Ezt a kérdést az ún. „indító jellel" (azaz „aktív" üzemmódba indító jellel) oldják meg. Ezt már tárgyaltuk.

A másik fontos feladat több operátor egyidejű figyelése. Ennek több célja lehet. Előfordulhat, hogy annak a jelzésére van szükségünk egy másik operátor számára, hogy a figyelt operátorok még mind valamilyen (például „aktív") üzemmódban dolgoznak. Másik gyakori feladat annak figyelése, hogy mind befejezte-e valamilyen jellegű működését, például mind „készen van"-e, azaz kilépett-e például az „aktív" állapotból. Máskor meg azt kell jelezni, hogy a megfigyeltek közül legalább egy üzemmódot váltott (például „készen van").

Fontos fogalom az üzemmód-nyilvántartás, az üzemmód-indikálás. Ezt a műveletet önálló operátorokkal is megoldhatjuk, de beépíthetjük abba az operátorba magába is, amelynek üzemmódjáról információt kell eljuttatnunk valahová.

„Indító" bemenettel (azaz „indító jelet" érzékelő karral) rendelkező operátorok esetében az „indító jel" „bekapcsolja" az üzemmód-indikátort, azaz olyan állapotba hozza kimeneti változóját (adó kezében olyan információs állapotot tart fenn), amely a szóban forgó üzemmódról szolgáltat egyértelmű információt. A „kész" jel pedig „kikapcsolja" az üzemmód-indikátort, azaz kimeneti változója értékét úgy választja meg, hogy ez az új állapotnak megfelelő legyen. Ezt a két funkciót – természetesen – beépíthetjük az operátorba magába is. Példánk esetében ez bizonyos értelemben vett foglaltságról fog tudósítani. Ilyenkor nyilvánvalóan eggyel több adó karja lesz operátorunknak.

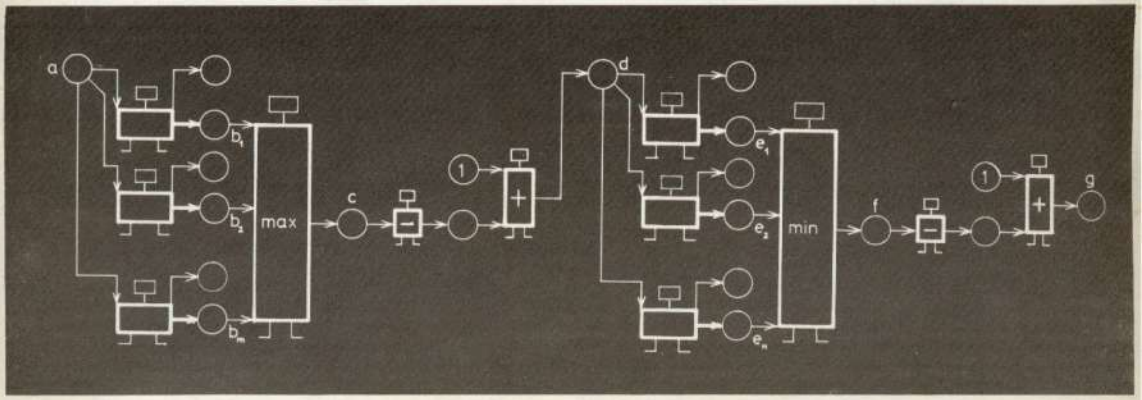
Formailag teljesen hasonló, de tartalmilag nem szükségképpen azonos a foglaltságindikátor (a foglaltságról tudósító adó kéz) szerepe a „működésben tartó" kapó kéz szerepével. Az olyan esetekben, amelyekben nincs tartalmi különbség a kettő között, egyik a másikkal – alkalmas robotkonstrukciókkal – helyettesíthető.

Szemléltető példa

Lássunk most egy egyszerű, de tipikus és gyakori működésmód-vezérlési példát! Két csoport operátorunk van. Az első csoport elemeit egyszerre hozzuk „aktív" üzemállapotba, egy impulzus felfutó élével. Az ilyen üzemmódra minden operátornak van valamilyen, véges időt igénylő feladata, és előírt feladatát mindegyiknek el kell végeznie. Operátoraink olyanok, hogy a „foglalt", azaz aktív üzemmódban kapott indítójelek „hatástalanok". Ha feladatával mindegyik operátor elkészült, azaz már egynek az üzemállapota sem „aktív", akkor „indítjuk", mégpedig késedelem nélkül és egyszerre a többi, a második csoportbeli operátort, ugyancsak egy felfutó éllel. Figyelnünk kell az operátorok között a „legfürgébbet", és a legkorábbi feladatelvégzést, az „aktív" üzemmódból való legkorábbi kilépést felfutó éllel jeleznünk kell.

A feladat megoldására két gyakorlati út van.

1. ábra



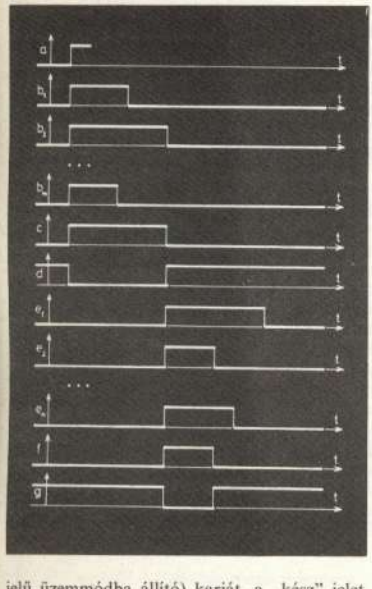
Az egyik akkor alkalmazható, ha előre tudjuk az operátoraink feladata elvégzéséhez szükséges időket. Ha ezek az első csoportban p_1, p_2, ... p_m, a másodikban pedig q_1, q_2, ... q_n, akkor semmi külön megfigyelésre nincs szükségünk. Az „indító jel" után a kívánt jelet

$$\max(p_1; p_2; \ldots; p_m) + \min(q_1; q_2; \ldots q_n)$$

idővel kell adni. Ha viszont a feladatvégzési időket nem ismerjük előre, elkerülhetetlenül figyelnünk kell minden egyes operátorunk működési módját. Legyenek operátoraink olyanok, hogy van működésmódot jelző, indikáló kezük is. A „foglalt", az „aktív" állapotot (működésmódot) jelezze 1, a „szabad"-ot pedig 0, az operátor állapotjelző „kezében".

Operátorainknak csak az „indító" (azaz 1 jelű üzemmódba állító) karját, a „kész" jelet adó (azaz az 1 jelű működési állapotból a 0 jelűbe való átmenetet jelző) karját és az üzemmódot jelző indikáló karját ábrázoljuk (ez utóbbit vastag vonallal).

2. ábra



Az operátorcsapat „felállása", azaz a kívánt feladatot megoldó rendszer konstrukciója az *1. ábrán* látható.

A *2. ábra* a rendszer működéstörténetét leíró folyamatokat tartalmazza.

Az 1. ábra sok rokon vonást mutat az Alapozás III. 2. ábrájával. Ez nem váratlan. Van azonban egy lényeges eltérés is, mégpedig az, hogy a III. rész 2. ábráján egy operátort több más operátor is „indíthat" (azaz üzemmódját több más operátor is befolyásolhatja). Erre nyilvánvalóan lehetőséget kell biztosítani, de az is igaz, hogy nem tudjuk megmondani, hogy a **let c** operátor értékadási tevékenységét indító változójának értékét melyik operátor határozza meg. Mivel minden operátor minden kimeneti változójának értéke meghatározott, két különböző operátor egy-egy kimeneti változójának egyesítése csak akkor jogos, ha a két szóban forgó változó alakulása az időben teljesen azonos. Ez viszont esetünkben nem így van.

Mi tehát a helyes megoldás? A III. részben az indítási folyamat részleteivel még nem foglalkozhattunk. A III. rész 2. ábrája ezért csupán elveket rögzít, nem tekinthető egzakt kapcsolatábrának. Miután konkrétan meghatároztuk az „indítás" (az állapotváltás) lefolyását, a kérdés megoldhatóvá válik. Előbb nem, hiszen más-más „indítási" folyamat más-más kezelést igényelhet. Ha az „indítás" egy rövid impulzus felfutó élével történik, a **let c** operátor „indító" bemenetét egy olyan közbeiktatott összegképző vagy maximumképző operátorral működtethetjük, amelynek nincs két üzemmódja, hanem csak egy; ugyanis ezeknek állandóan működniük kell. Így tehát ezek esetében nincs értelme üzemmódbeállító, üzemmódváltoztató bemeneti változókról beszélni. Az összegképző, illetve a maximumképző operátor két bemeneti kezе fogadja a két különböző helyről jövő „indító" jelet.

2. ábránk maximumképzője, minimumképzője, ellentettképzői és összegképzői szintén állandóan egyféle üzemmódban működnek.

Összefoglalás

Először is egy taktikai figyelmeztetés. Ha valakinek esetleg egyik-másik részlet első olvasásra túlzottan elvontnak tűnik, ne keseredjen el. Ha a szemléltető példát érti, és önállóan, puskázás nélkül le tudja rajzolni az operátorcsapat felállását, akkor érti a lényeget, és teherbíró alappal rendelkezik a további tudásanyag számára. Később azonban vissza-vissza kell térnie az átugrott részekhez, addig, amíg minden részlet természetes és könnyen érthető nem lesz.

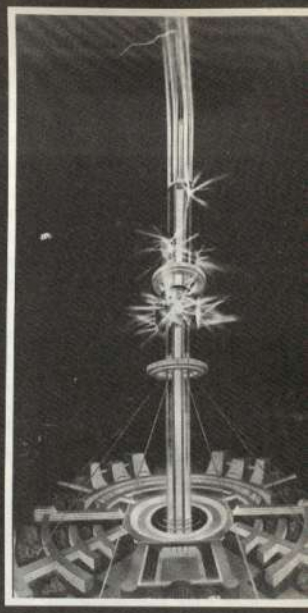
Másodszor arról ejtsünk néhány szót, hogy hol is állunk most tudásalapozási munkánkban. Meg kell mondanunk, hogy a nehezén túl vagyunk. Minden számítástechnikai folyamat lényege, lelke olyan jellegű, mint amilyet szemléltető példánk bemutat. Ilyen és csak ilyen jellegű folyamatok fordulhatnak elő, akár a szgépet alkotó elemi gépi operátorok (alkatrészek), akár programok, részprogramok között. A megértés kulcsa tehát már a kezünkben van. Az elméleti alapok leglényegesebb elemei már a birtokunkban vannak.

Következnek a gyakorlati megvalósítások lehetséges útjai. Ehhez azonban meg kell mondani, hogy az eddig elmondottak a gyakorlatban ilyen tisztaságban sohasem fordulhatnak elő. A felfutó és lefutó él például sohasem merőleges az időtengelyre. Ez viszont több érdekes kérdést is felvet. Néhány példa:

Egy „ferdén" felfutó él melyik időpontját kell „indító" időpontnak tekinteni? A felső csúcsát? Vagy a közepét? A fel- és lefutó élek meredeksége mindig ugyanaz? Van-e a meredekségnek határa, aminél laposabb jel például már hatástalan?

Ezeket a kérdéseket is tisztázni fogjuk a következőkben.

POGÁNY CSABA



A Nap tornya (terv: Montreál)

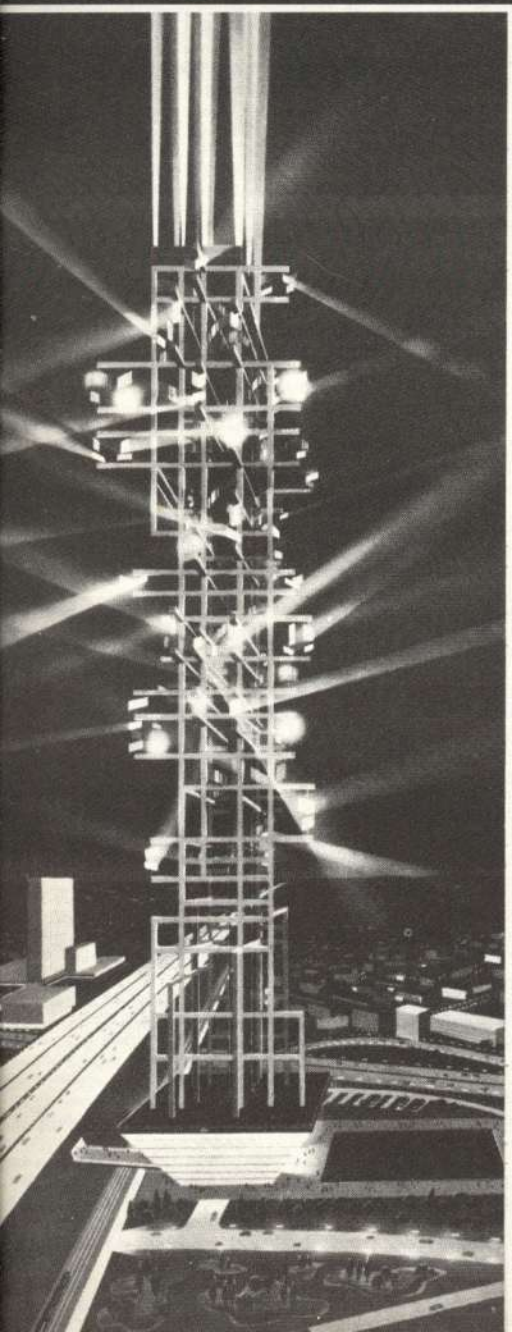
Nicolas Schöffer 1912-ben született Kalocsán, 1936-tól Párizsban él. A francia Népművészeti Akadémia tagja. Szobrász, de építész is, zeneszerző is. Leginkább egyetemes művész, aki saját megfogalmazása szerint műveit olyan „anyag nélküli anyagokkal" komponálja, mint a tér, a fény és az idő. Műveivel már korábban találkoztam, a művésszel először 1983-ban, Párizsban. Ő az első, akinek alkotásaiban lényeges szerepet kapott a számítástechnika „anyaga", az információ.

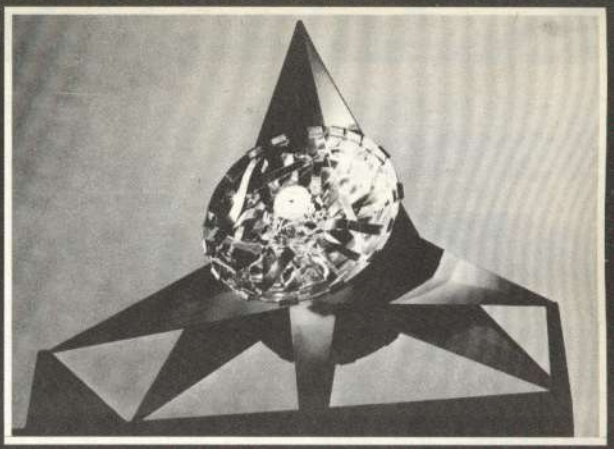
1948-ban dolgozta ki térdinamikai elképzeléseit, és 1950-ben állította ki térdinamikai szobrait Párizsban. Az első kibernetikus szobrát is Párizsban mutatta be, 1956-ban. A szobrok fénnyel, hanggal és mozgással keltenek hatást, ami a korai alkotásokban elektromechanikus, az újabbakban számítógépes vezérlésű.

A szobrok programját olyan véletlen környezeti jelenségek változtatják, mint a hang, a fény, a légmozgás, a nedvesség stb. A szobor viselkedését ún. programozott zavarok kibillenthetik az eredeti programból, ezután új egyensúlyi állapotban, új módon válaszol a környezet jelzéseire.

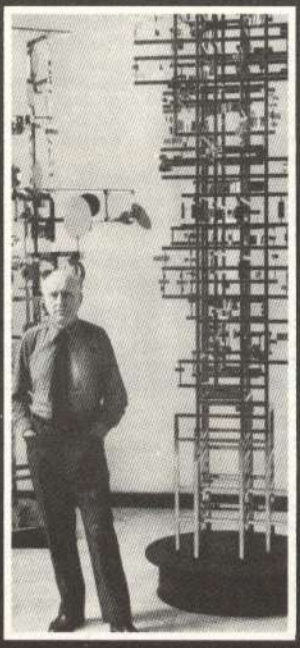
Nicolas Schöffer jellegzetes alkotásai a kibernetikus fénytornyok. Véleménye szerint az egyik legjobb, vagy talán a legjobb alkotása Kalocsán, szülővárosában áll.

NICOLAS SCHÖFFER





Prizma
(Szépművészeti Múzeum)



A művész

Monumentális
kibernetikus fénytorony

Németh Lajos Schöffer 1982. évi budapesti kiállítása katalógusában ezt írja: „Schöffer kibernetikus fénytornyai, nagyszabású programot teljesítő 'Chronos'-ai, lézerkompozíciói a kibernetikus logika, a vezérelt program, az elektronika szabályát, tökéletes működését követő mechanizmusok, ugyanakkor a szüntelen mozgásnak, 'alakváltozásnak', a környezetet éltető, mozgató ritmusuk révén a változás, a véletlen, az élet lüktetésének szimbólumai is: a rend és a rendetlenség erőegyensúlyának képletei. Schöffer művei korunk technikai forradalmának, a kibernetikának és elektronikának a talaján születtek, ugyanakkor minden újszerűségük ellenére kapcsolódnak a művészet nagy hagyományaihoz is. Hiszen művészetének alapja a fény, a tér dinamikája, a mozgás, az alakulás, a változás."

Schöffer kibernetikus fénytornyai a világ sok nagyvárosában (sajnos Budapesten még nem) jelzik a tér és az idő változását: San Franciscóban, Bonnban és természetesen Párizsban. Két legnagyobb terve egyelőre még terv maradt: egy Los Angeles-i és egy párizsi óriási fénytorony, amelyek egy-egy korszakának összegzését adták volna. A budapesti Szépművészeti Múzeumnak a művész egy prizmát ajándékozott, amelyben tükrök sokszorozzák a prizma középpontjába állított kis fénytorony mozgását és fényeffektusait.

Nicolas Schöffer sokoldalúságát a hangstruktúrákról végzett tanulmányai is jelzik. Legutóbbi találkozásunk alkalmával meghallgathattam „Variations sur 600" című hangstrukturális kompozícióját, amelyet a Pompidou Center 4× számítógépen alkotott. A kompozíció leírására nem vállalkozom, nem értek hozzá. Egyetlen érdekes dolgot emelek ki. Nicolas Schöffer a művet a szoborkompozícióhoz hasonló módszerrel számította ki és programozta; azt is mondhatnám, hogy a muzsika szobrainak egy más közegben létrehozott absztrakciójának tekinthető.

Azt hiszem, nem kell különösen magyaráznom, hogy mi, a számítástechnika művelői, Nicolas Schöfferben nemcsak a művészt becsüljük, de azt a honfitársunkat is, aki a kibernetikát és a számítástechnikát talán elsőként alkalmazta művészi gondolatainak kifejezésére.

K. GY.

Ismét a FORTH-ról

Mielőtt a kedves olvasó gyanakodva kapná fel a fejét, mondván: csak nem változott át vitafórummá a Programozástechnika című rovat, sietve leszögezem a következőket.

Magam is lelkes, mondhatnám elkötelezett híve vagyok a FORTH programozási rendszernek, és a számítástechnika alkalmazásának jelentős megnövekedését, a számítástechnikai kultúra jelentős megerősödését várom elterjedésétől. Természetesen elsősorban a mikroszámítógépek (személyi számítógépek, iskolaszámítógépek) körében.

Pataki Ernő cikkét (μM 1983. évi szám) kitűnő bevezető ismertetőnek tartom, amit az is igazol, hogy a cikk megjelenése óta az érdeklődés ugrásszerűen megnőtt. Éppen ez a megnövekedett érdeklődés indokolja, hogy e folyóirat hasábjain ismét foglalkozzunk a FORTH-szal.

Miután nem feltételezhetem, hogy Pataki Ernő cikkét mindenki olvasta, ugyanakkor fölösleges ismétlésekkel sem kívánom terhelni az olvasót, az alábbi megoldást választottam.

A cikk első részében egy új szemléletmódot követve ismertetem a FORTH rendszert. Ennek az a lényege, hogy gondolatban megkonstruálunk egy FORTH számítógépet (remélem, az olvasó aktív „szellemi" részvételével). A második részben megvizsgáljuk ennek a FORTH gépnek legfontosabb tulajdonságait, nem elhallgatva esetleges hiányosságait sem. Ezután ismertetem a FORTH-szal foglalkozó irodalom legfontosabb közleményeit. Végül a FORTH magyarországi múltját, jelenét és javasolt jövőjét tárgyalom.

Készítsünk tehát egy FORTH számítógépet!

Mielőtt a tervezéshez hozzákezdünk – ha csak gondolatban is –, két dolgot meg kell határoznunk: mi a konstrukció célja, és milyen, már meglévő környezetre támaszkodhat.

Az első kérdésre egyelőre csak roppant egyszerű választ adunk: célunk olyan programozási eszköz létrehozása, amely az adott alkalmazáshoz (ti. amire a programot írjuk) rugalmasan alakítható, és amely párbeszédes programfejlesztést tesz lehetővé.

A második kérdésre konkrétabb a válasz: feltételezünk egy számítógépet gépi utasításrendszerével, legalább 16 kbájt memóriával, valamint egy alfanumerikus terminált és valamilyen háttértárolót (például hajlékonylemezt).

Minden számítógépre jellemző az utasításkészlete. FORTH-ban az utasítást *szónak* nevezzük. Egy szó formailag bármilyen karaktersorozat lehet (nemcsak betű és szám), amelyet szóköz zár le (és választ el a következő szótól). Miután rugalmasan alakítható berendezést konstruálunk, nem alakítunk ki rögzített szókészletet, hanem lehetővé tesszük új szavak definiálását.

Egy új szó elvileg kétféle módon definiálható: már definiált szavak felhasználásával és az adott (valódi) gép gépi utasításrendszerének segítségével.

Fentiek formája legyen

: újszó szó1 szó2 ... ;

illetve

CODE újszó gépiut1 gépiut2 ... ENDCODE

(Vegyük észre, hogy a „:", a „;", a „CODE" és az „ENDCODE" is szavak, bár több szempontból is speciálisak, amiről később még lesz szó.)

Ha egy szó már definiálva van, akkor végrehajtható. A végrehajtást úgy kezdeményezhetjük, hogy az adott szót begépeljük a terminálon.

Végrehajtható műveleteink már vannak, gondoskodnunk kell ezek paraméterezéséről. Hozzunk létre FORTH gépünkön egy ún. adatvermet, és állapodjunk meg abban a konvencióban, hogy a szavak bemenő paramétereiket az adatveremről olvassák le (elmozgatván onnan azokat), majd az eredményt is az adatveremre írják rá.

Meg kell még oldanunk azt a problémát, hogy amikor egy szót végrehajtásra begépelünk, hogyan juttatjuk el a bemenő paramétereket a verembe. A következő megoldást választjuk: Ha a terminálon begépelünk valamit, a rendszer megpróbálja azt először szónak értelmezni. Ha ilyen szót nem talál, megpróbálja számnak értelmezni azt; ha ez sikerül, a számot beírja a verembe. Az eredmény kinyomtatására külön szót definiálunk, a „."-t. Ha ez szerepel az adott szó definíciójában, az eredmény kinyomtatódik, ellenkező esetben nem.

Természetesen ha egy szót már definiáltunk, ez nemcsak arra használható, hogy végrehajtsuk, hanem szerepelhet új szavak definíciójában is. A definícióban szereplő szavak paramétereiket szintén az adatvermen keresztül adják át egymásnak.

Például gépeljük be ezt:

: PLUS + . ;

(Tételezzük fel, hogy a „+" szó már definiálva van, két számot vár a vermen, és eredményként azok összegét rakja a veremre.)

5 6 PLUS

Ennek hatására a 11 kinyomtatódik a terminálon. Könnyen belátható, hogy a fentiekben leírt gép már alkalmas programozásra, méghozzá például a következő módon. Először definiáljuk a legalapvetőbb műveleteket a CODE ENDCODE szópár segítségével (ezeket primér szavaknak nevezzük). Ezután bonyolultabb operációkat végrehajtó sorokat definiálunk a már definiált szavakkal (szekundér szavak). Aztán részprogramokat, majd végül az egész programot definiáljuk egy-egy szóval. Most már a program nevét jelentő szó begépelésével a program végrehajtható. Létrehoztuk tehát egy már működő gép fekete dobozát.

Nézzünk bele a FORTH gép fekete dobozába!

Azt már tudjuk, hogy a gép tartalmaz egy adatvermet a szavak ki- és bemenő paramétereinek tárolására. Miután a gépet terminálról működtetjük, szükség van a bemenő mondat tárolására (ez az a szó vagy szavak, amit begépelünk), erre szolgál az inputpuffer. De felmerül a kérdés, hogy hol és hogyan tároljuk a már definiált szavakat. És most elérkeztünk a FORTH gép legfontosabb tárolóegységéhez, a *szótárhoz* (dictionary). A szótár szótári elemekből áll, minden definiált szóhoz tartozik egy elem.

Egy ilyen szótári elemnek kétféle információt kell tartalmaznia: egy olyat, amelynek segítségével a szó a szótárban megkereshető, nevezzük ezt a szótári elem *fejének*, továbbá egy olyat, amelynek segítségével a szótári elemhez tartozó szót a gépünk végre tudja hajtani. (Ezeket az információkat a szó definiálásával adtuk meg.) Nevezzük ezt az elem *törzsének*.

Mind a fej, mind a törzs további két részre bontható. A fej tartalmazza természetesen a szó nevét a *névmezőben* és egy *kapcsolómezőt*, amely lehetővé teszi, hogy szekvenciálisan nem egymás után következő szavakat logikailag összekapcsoljunk. (Ennek szerepéről később lesz szó.)

A törzs adatainak interpretálásával kell az adott szót végrehajtani. Miután másképpen kell interpretálni például a „:"-tal és másképpen a „CODE"-dal definiált szót, a törzsnek tartalmaznia kell az interpretálás módjára – pontosabban az interpretálást végző segédprogram kezdőcímére – vonatkozó információt. Erre szolgál a *kódcímmező*. Az interpretálás tárgyát képező adatokat a *paramétermező* tartalmazza.

A paramétermező a „:"-tal definiált szavak esetén a definiálásban részt vevő „régi" szavak címeit, a „CODE"-dal definiált szavak esetén pedig magukat a gépi utasításokat tartalmazza.

Lássuk ezután, hogyan működik FORTH gépünk.

A begépelt szó vagy szavak az inputpufferbe kerülnek. Ezután a külső interpreter (parancsinterpreter) kapja meg a vezérlést. Ez megkeresi a begépelt szót a szótárban, majd a kódcímező alapján elindítja az aktuális interpretáló segédprogramot. Ha a paramétermezőben „régi" szavak címei vannak, („:"-tal definiált szó), akkor ez a segédprogram a belső interpreter (FORTH interpreter), amely most már a szavak címei alapján keresi meg (direkt elérés) és hajtja végre azokat. Ez a végrehajtás többszörösen egymásba ágazott lehet – azaz több szekunder szón keresztül jutunk el a primer szóig. Ehhez ismét szükségünk van egy veremre. FORTH gépünk tehát még egy vermet fog tartalmazni, amelyet nevezzünk *programveremnek*.

Primér szó végrehajtása lényegesen egyszerűbb. A kódcímmező a paramétermező kezdőcímét tartalmazza. Így a gép rátér a paramétermező végrehajtására. Egyetlen speciális utasításra van csak szükségünk, a NEXT utasításra, amely a paramétermező utolsó utasítása, és amelynek hatására visszatérünk vagy a külső vagy a belső interpreterbe (attól függően, hogy melyik aktivizálta az adott primer szót).

Már említettük, hogy a „:" és a „CODE" is szavak, tehát a szótár ezeket is tartalmazza. Mindamellett ezeket határozottan meg kell különböztetnünk azoktól a szavaktól, amelyeket éppen az ő segítségükkel definiáltunk.

Mi ez a különbség? Mind a két szó definiáló szó, azaz végrehajtásuknak az az eredménye, hogy létrehoznak egy szótári elemet. FORTH gépünkbe természetesen az említett két definiáló szón kívül még másokat is beépítünk. Itt most két további nagyon fontos definiáló szót említünk meg: a VARIABLE szó segítségével egy változót, a CONSTANT szó segítségével pedig egy konstanst lehet a szótárba felvenni. Ezek formája:

VARIABLE név
CONSTANT név

A VARIABLE hatására a szótárban a névmezőbe a név, a paramétermezőbe egy „üres" memóriahely kerül. Ha a VARIABLE-lal definiált szót végrehajtjuk, az adatveremre a változó címe (a szótárbeli paramétermező címe) kerül.

A CONSTANT szó egy értéket vár a veremben, és hatására ez az érték kerül a paramétermezőbe. A definiált név végrehajtásakor a konstans értéke kerül a verembe.

Itt álljunk meg egy pillanatra. Említettük, hogy nem határozunk meg fix szókészletet, hanem új szó definiálására adunk lehetőséget. A gyakorlatban persze ez nem egészen így van. Hiszen például egy átlagos felhasználótól nem várható el, hogy a definiáló szavakat is maga definiálja, tehát ezeket mindenképpen be kell építenünk a gépbe. A gyakorlatban ezenkívül az összes alapvető aritmetikai, logikai, vezérlési, bemenő, kimenő szavakat és még egy sereg hasznos egyéb szót be szoktak építeni a FORTH gépbe. Egy tipikus FORTH gép mintegy 150–250 beépített szót tartalmaz.

Foglaljuk össze, hogy eddig megkonstruált FORTH gépünk milyen tároló- és programegységeket tartalmaz.

A tárolóegységek: szótár, inputpuffer, adatverem, programverem. A programegységek pedig: külső interpreter, belső interpreter, compiler és assembler. Az utóbbi kettőről még nem volt szó, ezért ezeket röviden ismertetem.

A compiler az a segédprogram, amire a „:" szó kódcímmezejének tartalma mutat. Feladata, hogy a definícióban szereplő „régi" szavak címeit megkeresse, és ezeket az „új" szó paramétermezőiben elhelyezze.

Az assembler az a segédprogram, amire a CODE szó kódcímmezeje mutat. A CODE szóban a gépi utasításokat egy speciális FORTH assembler formában adjuk meg, az assembler ezeket fordítja le gépi kódra, és helyezi el a definiálandó szó paramétermezejébe.

Fejlesszük tovább FORTH gépünket!

Mint az előzőekből kiderült, a definiáló szavak alapvető szerepet játszanak FORTH gépünk működésében, ezért ezeket a „legminimálisabb kiépítésű" FORTH gépbe is eleve be kell építenünk. De a definiáló szavak is csak szavak, ezért jogos igényként merül fel – legalábbis a hivatásos számítástechnikusok részéről –, hogy új definiáló szavakat is definiálhassanak, méghozzá szekunder szóként is. Ez nagyon fontos szolgáltatás, mert lehetővé teszi a FORTH nyelv valódi kiterjesztését, miután így nemcsak új műveleteket, hanem új adat- és vezérlőstruktúrákat is lehet definiálni.

Ehhez gondoljuk meg, hogy mit kell egy definiáló szó definiálásakor megadni: azt, hogy a definiálandó definiáló szó mit és hogyan rakjon be a szótárba (fordítási idő alatti viselkedés, compile time behaviour), továbbá, hogy miután a definiált definiáló szóval definiáltunk egy szót, és ezt végrehajtjuk, mi történjen (futási idő alatti viselkedés, run time behaviour).

Mindehhez a FORTH a következő formulát adja:

: „definiáló szó neve" CREATE „fordítási idő alatti viselkedés" < DOES „futási idő alatti viselkedés";
Példa:
: CONSTANT CREATE , DOES > @ ;

Fordítási idő alatt a „," szó hatására a konstans értéke bekerül a paramétermezőbe, futási idő alatt „@" szó hatására az adatverembe.

Egy másik, az előzőnél talán még nehezebb probléma az ún. összetett vagy strukturált utasítások. A FORTH alapgép tartalmaz néhány ilyet. Például:

IF ELSE ENDIF

a feltételes elágazás utasítása.

Természetesen ez esetben is biztosítanunk kell, hogy a felhasználó is létrehozhasson ilyen utasításszerkezeteket.

Hogyan oldható meg a fenti probléma anélkül, hogy a FORTH legalapvetőbb építőkövét, a szó koncepciót megsértenénk?

A megoldás amilyen zseniális, olyan egyszerű is. Vegyünk most egy másik példát, a ciklus utasítást, amely így néz ki:

„határ" , „kezdőérték" DO „ciklusmag" LOOP

Természetesen mind a DO-t, mind a LOOP-ot különálló szónak tekintjük, csak megteremtjük a megfelelő kapcsolatot közöttük. Ehhez a compilert kell kibővíteni. A compiler feladata eddig csak az volt, hogy ha megtalált egy szót, annak a címét berakta a paramétermezőbe. Most ezt a feladatot kell kibővíteni. A DO szó esetén például fel kell jegyeznie azt a címet, amin a DO van, hogy majd a későbbi LOOP szó egy olyan szót tudjon lefordítani, amely végrehajtáskor erre a címre ugrik vissza. De a DO-nak ezt a tevékenységét is a FORTH-ban kell megírni, azaz erre a tevékenységre is definiálnunk kell egy szót. Ezt a szót azonban a compilernek (amikor egy definícióban megtalálja), nem lefordítania kell (azaz a címét beraknia a szótárba), hanem végrehajtania.

A megoldás tehát kézenfekvő. Lehetőséget kell adnunk arra, hogy amikor egy szót definiálunk, deklarálhassuk, hogy az adott szó fordításkor nem fordítandó, hanem végrehajtandó. Ezt a deklarációt az IMMEDIATE szóval végezhetjük el.

Itt most befejezzük FORTH gépünk konstruálását. Az olvasónak két dologgal kapcsolatban lehet hiányérzete: nem foglalkoztunk a szövegkezeléssel és a háttértár-kezeléssel. Természetesen a FORTH programozási rendszerek ezt a két problémát is megoldják. Ezek tárgyalása azonban meghaladná e cikk kereteit, és végül is a „FORTH eszméhez" – amelynek megismertetése volt fő célunk –, nem tenne hozzá túl sokat.

Mit ad a FORTH, és mit kell érte fizetnünk?

Interaktivitás, megbízhatóság

Párbeszédes programozást a FORTH-on kívül számos programnyelv is lehetővé tesz (BASIC, APL, FOCAL stb.). A FORTH esetében ez különleges jelentőséget nyer. Ugyanis mivel a szavak viszonylag függetlenek és önállóan is végrehajthatók, lehetővé válik, hogy egy szót megírása után rögtön letesztelünk. Így mindig csak egy viszonylag kis programrészre kell koncentrálnunk, ami a szokásosnál lényegesen megbízhatóbb programozást tesz lehetővé.

A gyakorlatban azért nem ilyen egyszerű a dolog. Tételezzük fel, hogy egy sokszorosan szekunder szót tesztelünk. (Ezen azt értjük, hogy a szó definíciójában szerepelnek olyan szavak, amelyek végrehajtásánál csak nagyon sok lépcsőben lehet eljutni a primer szóig.) Továbbá feltételezzük azt is, hogy az összes alsóbb szintű szót korrektül letesztelték, valamint azt, hogy magában a szóban, amit tesztelünk, nem találunk hibát. A tesztelés eredménye mégis lehet negatív. Ennek az lehet az oka, hogy a szó definíciójában szereplő valamelyik szót nem helyesen használunk, azaz nem úgy, ahogy azt a tervezéskor elképzeltük, és amire a szót letesztelték. Ebben az esetben a hiba kiderítéséhez esetleg le kell mennünk a primer szintig. Miután egy tipikus FORTH program több száz szót is tartalmazhat, az ilyenfajta hibát nagyon könnyű elkövetni.

A hiba kijavítására nagyon hasznos segédeszközt ajánlhatunk. Minden definiált szóhoz egy rövid leírást (glossary) lehet készíteni. Ez tartalmazza a szó funkciójának leírását egy-két mondatban, valamint a szó hatását az adatveremre az alábbi formula szerint:

$(a_1 a_2 \ldots a_n - - - - b_1 b_2 \ldots b_m)$

A szó $a_1 a_2 \ldots a_n$ adatokat várja a vermen végrehajtás előtt, és $b_1 b_2 \ldots b_m$ adatokat hagyja a vermen végrehajtás után. a_n, ill. b_m a verem tetején levő elem. Ezeket a leírásokat is célszerű a lemezen tartani, lehetőleg abc sorrendben.

Tömörség, fej nélküli kód

A FORTH-ban írt programok a szokásosnál jóval kevesebb tárt igényelnek. Már működő, véglegesnek ítélt programokban ez a tömörség még fokozható úgy, hogy a szótárban a fejrészt elhagyjuk (hiszen a futó program már címek alapján keresi meg a szavakat, nem név szerint). A tömörség még tovább fokozható azáltal,

hogy a FORTH alapszókészletéből teljes egészében kihagyjuk azokat, amelyeket az adott program nem használ. Persze e módszerek alkalmazásához célszerű segédprogramokat készíteni. Természetesen ezek a programok is megírhatók FORTH-ban. Az említett módszereket elsősorban célgépek (például robotok) vezérlő programjainak előállításához célszerű használni.

Szabványosítás, hordozhatóság

Már többször említettük, hogy a FORTH rendszerek eleve tartalmaznak egy kb. 150–250 szóból álló alapkészletet. Miután egy szó definiálása elég egyszerű és a számítástechnikusok általában „fantáziadús" emberek, minden egyes létrehozott FORTH rendszer alapszókészlete más és más. (Jelentős magyarországi FORTH alkalmazás a SZÁMOK által kifejlesztett mikrogépes adatbázis-kezelő rendszer.)

Hogyan értik meg mégis egymást a különböző FORTH szakemberek, egyáltalán hogyan lehet akkor FORTH programokat publikálni?

A probléma megoldására született meg 1979-ben a FORTH–79 szabvány. Ez meghatároz kb. 150 alapszót, amelyet minden egyes FORTH implementációnak tartalmaznia kell. Ezenkívül javaslatot tesz mintegy 100 ajánlott szóra.

A „FORTH körökben" kialakultak bizonyos elnevezési konvenciók is. Például mint írtuk, a „." az adatverem legfelső elemének kinyomtatását jelenti. Így például, ha egy hibaszöveget nyomtató szót akarunk definiálni, akkor ez a konvenciók szerint valami ilyesmi lesz:

.ERROR

Magas szintű nyelvek esetén általában a szabvány a hordozhatóság eszköze. Tehát, ha mindenki a szabványnak megfelelően írná FORTRAN programjait, akkor ezeket a programokat probléma nélkül át lehetne vinni egyik gépről a másikra. Akinek akár csak pár év gyakorlata van a számítástechnikában, az tudja, hogyan működik ez a dolog. (A számítástechnikában gyakorlatlanok számára azért elárulјuk, hogy gyakorlatilag sehogy.)

A hordozhatóságot (mint általában a legtöbb dolgot az életben) adminisztratíve – tehát szabványos előírásokkal – nem is lehet megoldani. Nem a szabvány szükségességét tagadjuk, hanem azt, hogy elégséges is a probléma megoldására.

A FORTH nyelven írt programok nagyfokú hordozhatósága nem is a FORTH–79 szabványnak, hanem a nyelv talán legfontosabb tulajdonságának, *nyitottságának* köszönhető. Tehát annak, hogy új műveletek, adat- és vezérlési struktúrák definiálása, illetve a régiek átdefiniálása a FORTH-ban triviális és gyorsan végrehajtható feladat.

Ha tehát egy FORTH-ban írt programot át akarunk vinni az egyik FORTH rendszerről a másikra, akkor a következőt tehetjük. Először megállapítjuk, hogy a két rendszer alapszókészlete között mi a különbség, majd új szavak definiálásával, illetve a régiek átdefiniálásával ezt a különbséget megszüntetjük. Ezután a program már átvihető egyik gépről a másikra. Ezt a feladatot általában 1–7 nap alatt el lehet végezni.

Mit olvashatunk a FORTH-ról?

A bevezetésben említett Pataki Ernő cikken kívül magyar nyelvű publikáció nincs. Angol nyelven eléggé kiterjedt irodalma van a FORTH-nak; ebből most csak a legfontosabbakra térek ki.

Úgy hirdetik, hogy a legjobb tankönyv e témában Leo Brody: Starting FORTH című könyve. Hajlamos vagyok azt hinni, hogy a hirdetéseknek ezúttal teljes mértékben igazuk van. Ez a könyv mind teljesen kezdőknek, mind hivatásosoknak nagyszerű olvasmány. Elég vastag (kb. 300 oldal), de ha az ember végigolvassa, mindenképpen FORTH programozónak tekintheti magát.

Egy másik alapvető jelentőségű könyv R. G. Loeliger Threaded Interpretive Languages című munkája. Azoknak ajánlom, akik mélyebb ismereteket kívánnak szerezni a FORTH-ról, elsősorban implementációs kérdésekben.

Végül megemlítem a FORTH Dimensions című folyóiratot. Amerikában létrejött a FORTH felhasználóknak egy egyesülése, a FIG (FORTH Interest Group). Ennek negyedévenként megjelenő hivatalos folyóiratáról van szó, amely elsősorban új FORTH programokat közöl.

Mi a helyzet FORTH-ügyben Magyarországon?

Mint a bevezetőben is említettem, a FORTH egyre jobban terjed hazánkban. Bár még a FORTH korszak elején járunk, véleményem szerint népszerűségben máris túlhaladtuk a nyugateurópai szintet. Szerintem ebben három tényező játszik szerepet: a mikroszámítógépek elterjedése hazánkban is; a számítástechnikai kisvállalkozások elszaporodása; a szoftvertermék-piac gyengesége, a nyugati termékek nehéz beszerezhetősége.

Az első tényezőhöz nincs mit hozzáfűzni, a FORTH tipikusan mikroszámítógépes rendszer, és általában elég olcsón beszerezhető (Nyugaton 50–200 dollár között, Magyarországon 10 ezer–30 ezer forint között).

Úgy tűnik, hogy a FORTH az ideális eszköz a kisvállalkozások számára. Elég gyorsan készíthetők vele ugyanis hatékony, jól dokumentált és magas szinten karbantartható rendszerek.

Miután hazánkban a piacon viszonylag kevés az univerzális, jól használható alkalmazói programcsomag, jóval fokozottabb az igény olyan eszközökre, amelyek a „csináld magad" módszert támogatják. Sajnos még mindig egyszerűbb, ha az ember maga csinál magának például egy adatkezelőt, mint ha megpróbálja megvenni azt (hacsak nincs dollárja). Meggyőződésem, hogy a „csináld magad" módszerhez pillanatnyilag nincs Magyarországon a FORTH-nál jobb eszköz.

Fentiek alapján megfontolandó, hogy nem kellene-e nálunk is megszervezni az FFK-t, a FORTH Felhasználók Körét. Talán egy FORTH folyóirat indítása sem lenne elképzelhetetlen. Nagyon lényegesnek tartom a program- és információcsere megszervezését mind jogi, mind gazdasági vonatkozásban.

FORGÁCS TAMÁS

Clive lovag és az óriás

A személyi számítógépek sűrűsége az Egyesült Királyságban a legnagyobb az egész világon: minden tizedik családnak van kisszámítógépe (Japánban minden huszonötödiknek, az USA-ban pedig minden ötvenediknek). Anglia jó helyezésében nem kis érdeme van Clive Sinclairnek. Először is új árkategóriát nyitott a 100 fonton aluli ZX80-nal; aztán ennek továbbfejlesztett változatát, a ZX81-et már 40 font alatt árusította, majd a Spectrummal letörte az árakat abban az igényesebb kategóriában is, amelyben addig három amerikai cég uralkodott: az Apple, a Commodore és a Tandy.

Clive Sinclair tehát igazán megérdemelte, hogy a királynő lovaggá üsse. E szertartás óta pedig már nem a „Mr. Sinclair", hanem a „Sir Clive" cím illeti meg.

Sir Clive nem pihent meg babérjain, hanem – ahogy vitéz lovaghoz illik – új csatába indult. Ez év márciusában jelent meg a piacon új konstrukciója, a Sinclair QL. (A két betű jelentése: quantum leap, azaz kvantumugrás.)

„Mindazt, amit az IBM PC tud, azt a QL is tudja, csak gyorsabban és jobban" – jelentette ki Sir Clive Sinclair az új gép sajtóbemutatóján. Önérzetes mondás, ha figyelembe vesszük, hogy a QL ára csak 399 font, az IBM PC-é pedig ennek több, mint ötszöröse. A lovag tehát kesztyűt dobott az óriásnak.

Milyen az új gép? Külalakra hasonlít a Commodore C–64-re: áramkörei a klaviatúrával egy dobozba vannak építve. Ellentétben a kisebb Sinclair gépekkel, nem fóliaklaviatúrát használ, hanem szabványos írógépbillentyűzetet, kiegészítve négy kurzormozgató és öt f-billentyűvel.

Főműve Motorola 68008 processzort használ. Ez 1 M-ig terjedő címtartományt szegmentálás nélkül fog át. Tehát csakugyan veri az IBM PC-t, amelynek Intel 8088 processzora csak 64 k-s szegmentekben tud közvetlenül címezni. Az utóbbi hátránya főleg a fordítóprogramokban mutatkozik meg. A 8088-cal vagy 8086-tal ellátott gépeken akár 128 k, akár 256 k a főtár terjedelme, a BASIC mindenképpen azt közli a felhasználóval, hogy 60 k áll rendelkezésére saját programjai számára. Ezért Sir Clive-nek az a véleménye, hogy a következő négy-öt évben a 68000-es széria ki fogja szorítani vetélytársait (mint a nyolcbites processzorok között a Z80 tette).

A QL főtára alapkiépítésben 128 k terjedelmű; ez dugaszolható modullal 640 k-ra növelhető. Ebből 32 k-t a képmű bittérképe foglal el. A képmű egyébként lehet akár monitor, akár közönséges tévékészülék. Az előbbi esetben a sorhosszúság 85 karakter, az utóbbiban 40 vagy 60 karakter. A sorok száma mindkét esetben 25. Érdekes, hogy a karakterképek programozhatók, tehát például magyar vagy eszperantó ékezetes betűk is lehetnek. Grafikai felbontóképessége 512 × 256 képpont 8 színben (IBM PC: 640 × 200 képpont 5 színben).

A permanens tár kapacitása alapkiépítésben 32 k. Ez a Sinclair-Superbasic fordítót és a QDOS operációs rendszert tartalmazza. Dugaszolható modullal 64 k-ra bővíthető.

A Superbasic a kisebb Sinclair gépek BASIC-jének kiterjesztett, kompatibilis változata. Tudjuk azonban, hogy ezek is a más gépekétől nagyon elütő nyelvet használnak, tehát mondjuk az Apple vagy Commodore BASIC programok egyik Sinclair gépen sem futtathatók és viszont. A „BASIC" név használata tehát kapitalista reklámfogás.

A QL alapgépbe póttárként két különleges mikrokazettás készülék van beépítve, és kívül még hatot lehet csatlakoztatni. A kazetták végtelenített mágnesszalagot tartalmaznak. Kapacitásuk 150 k, átlagos elérési idejük 3,5 másodperc. A konkurrencia szakértői szerint az újmódi kazetták üzembiztossága és egymás közötti cserélhetősége a QL leggyöngébb pontja. Sir Clive azonban bízik a rendkívül olcsó új póttárban, és a kisebb Sinclair gépekhez is ezt akarja bevezetni. A jövő dönti el, kinek volt igaza.

Dugaszolható modul segítségével a Sinclair gépek hálózatba köthetők össze. Minden hálózat legfeljebb 64 gépet tartalmazhat: QL-eket és Spectrumokat vegyesen is.

A perifériacsatlakozók szabványosak (tehát nincs velük az a bosszúság, mint a Commodore gépekével, amelyek csak a saját nyomtatókkal köthetők össze). A QL soros csatlakozója az RS–232–C szabványnak felel meg; kapható külön párhuzamos csatlakozó is. Kurzormozgatásra két kormánybot csatlakoztatható.

A QL programkészletét a londoni Psion cég készíti. A gép ára (a permanens tárba beégetett programokon kívül) tartalmaz még egy szövegfeldolgozó, egy képfeldolgozó, egy rovatos számító (angolul spreadsheet) és egy adatbázis-kezelő programot. Ezekért tehát nem kell külön fizetni. További programok fejlesztése is folyamatban van. Minthogy a QDOS operációs rendszert külön a QL-hez fejlesztették, az eddigi két legnagyobb programkönyvtár: a CP/M-hez és az Apple-hez készült programoké nem használható a QL-hez. Erre érdemes lenne felfigyelniük a magyar programtervezőknek is. Mindazt, ami CP/M-hez vagy Apple-hez készen van, a QDOS-hoz vagy újra meg lehet írni, vagy adaptálni lehet. Nagy üzleti lehetőség!

A QL-et két cég: a Thorn EMI és a Timex gyártja. Egyelőre csak havi húszezer darabot, az év végére azonban havi százezerre akarják felfuttatni. Most még a brit piac mindent felvesz, kivált, mivel az ügyes Sir Clive szerződést kötött az ICL-lel, amely szerint ez a cég a QL-et fogja használni nagy gépei terminálisaként. Az exportot 1984 utolsó negyedében akarják megindítani.

MŰNNICH ANTAL

VME COOPERATIVE

Gyors gazdasági változások idején a technológiákat jóval gazdasági hatásuk előtt kell megtervezni, a viszonylag hosszú átfutási idő miatt. A pillanatnyi célszerűséghez való passzív alkalmazkodás helyett a nemzetközi tudomány által feltárt lehetőségeket kell megfontoltan, tudatosan kihasználni. Biztosítani kell, hogy az erőráfordítások elérjék legalább azt a minimális szintet, amely nélkül saját fejlődésre nem számíthatunk. Meg kell teremteni azokat a csoportokat, amelyek belátható időn belül elérik a kritikus fejlődési küszöböt. E csoportok együttműködési szerződésekkel, a fejlesztési stratégiák összehangolásával, a fejlesztők, a gyártók és a felhasználók új típusú együttműködésével válhatnak igazán hatékonnyá. Ennek szellemében kívánja az MTA SZTAKI létrehozni a COSY-VME KLUB-ot, amely közös érdekeltségi rendszert biztosít a tagok számára, akik így gazdaságosabbá és gyorsabbá tehetik alkalmazási feladataik megvalósítását az emberimunka-ráfordítások csökkentésével. A közösen kifejlesztett mikroszámítógépelemek (kártyák, perifériák, mechanikák, szoftver) modulokból épített, az egyedi fejlesztéseket könnyen befogadó rendszert képeznek.

A rendszer magja az elemek összekapcsolására alkalmas elektromos-mechanikai rendszer: a VMEbus. Napjainkban a világon már több mint 50 intézmény bocsát ki VMEbus kártyákat. A Motorola, a Mostek és a Signetics 1981 októberében javasolta szabványosításra az ISO-nak (International Standards Organization). A következő év augusztusában kiadták a részletesebb második változatot, majd 1983 márciusában az IEEE megalakította a szabványbizottságot. A VMEbus a multiprocesszoros rendszerek iránti igénynek köszönheti gyors sikerét, hiszen ezek jól átgondolt buszrendszer nélkül nem hozhatók létre. A 16 bites építőelemekkel egyidőben vált ismertté, de 8 bites elemekkel is jól használható. Lehetővé teszi 32 bites processzorok, memóriák, perifériák egységes rendszerben való kezelését.

A VMEbus négy jelvezetékcsoportot, illetve építőelemet és ezek együttműködésének szabályait határozza meg:

- adatátvitel
- vezérlőkijelölés
- megszakításkezelés
- szolgálati jelek

Az adatátviteli busz változtatható szélességű, adat és cím elkülönített (adat: 8–32, cím: 16–32) továbbítására alkalmas, 20 Mbájt/s sebességgel. Vezérlőjelei lehetővé teszik az ún. nem megszakítható read-modify-write üzemmódot is, amely többprocesszoros rendszerekben alapvető fontosságú. A címmódosító vezetékek blokkos átvitelt, memóriafelosztást, -hozzárendelést stb. tesznek lehetővé. A busz vezérlőjét, illetve használóját az első kártyahelyen lévő rendszervezérlő jelöli ki a kérelmek prioritása alapján. Ez a prioritás nem előre elhatározott, hanem az adatátvitel jellege szerint a master változtatja. A 7 szintű megszakítási kérelmeket bármely master megválaszolhatja. A gyors hibajelzést és a rendszer újraindítását biztosítják a szolgálati busz jelei. A rendszervezérlő kártyától függetlenül, önállóan működő soros busz a processzorok közötti kommunikációt szolgálhatja.

A VMEbus építőelemei EURO kártyán valósíthatók meg, egy, illetve két indirekt, 96 pontos csatlakozóval. A kártyák többsége kétszeres méretű (160 × 233,4), gyakran hosszított (220 × 233,4) kivitelű. A perifériák csatolása orrcsatlakozóval vagy a második kártyacsatlakozó egyes pontjain lehetséges. Az EUROCARD szabvány egyszerű mechanikai konstrukciót, az indirekt csatlakozók kiváló üzembiztonságot garantálnak.

A kezdetben SSI és MSI elemekből épített VME interfészek a Motorola LSI elemeivel kiválthatók lesznek. Gyártásban van a megszakításvezérlő (MC 68135) és a buszhasználat-vezérlő (arbiter, MC 68452). A Signetics 1984 II. félévére VMEbus-vezérlő kifejlesztését (68172), 1985-re pedig ennek hatékonyabb változatát ígéri (68173). A Motorola erre az időre új buszvezérlőt tervez (68174).

Az MTA SZTAKI által részben kifejlesztett, részben fejlesztés alatt álló VMEbus kártyák a szabvány alapján készülnek, 220 × 233,4 mm-es méretben. Az elemválaszték mechanikailag és elektromosan egységes, moduláris, szinte korlátlanul bővíthető mikroszámítógépek összeállítását biztosítja. Az alkalmazott aszinkron buszrendszer 8, 16 vagy 32 bites mikroszámítógép-elemeket fogad, címzési tartománya 16 vagy 24 bites lehet. A különböző sebességű, cím- és adattartományú elemek összekapcsolása nehézség nélkül elvégezhető. A VME Z80 VERSO hátlap tartalmaz Z80 buszokat is intelligens Z80 alapú perifériavezérlők számára. A perifériák a kártyákhoz orrcsatlakozón kapcsolódnak.

A BUSZRENDSZEREK ÖSSZEHASONLÍTÓ TÁBLÁZATA

	VMEbus	MULTIBUS	VERSABUS	Q-BUS	IEEE P896 „FUTUREBUS"
Típus	aszinkron, nem multiplexelt	aszinkron, nem multiplexelt	aszinkron, nem multiplexelt	aszinkron, multiplexelt	aszinkron, multiplexelt
16, 24, 32	20, 24	24, 32	16, 18	32	
Adatszélesség	8, 16, 32	8, 16	8, 16, 32	8, 16	8, 16, 32
Kártyaméret (mm)	160 × 100 vagy 160 × 233,4 vagy 220 × 233,4	171,5 × 304,8	255 × 368,3	214 × 263,4	160 × 100
Csatlakozótípusok	indirekt	direkt	direkt	direkt	indirekt
Csatlakozópontok száma	96/96	86/60	140/120	38	64 vagy 96
Tápfeszültség	+5, ±12, +5 Standby	+5, ±12	+5, ±12, ±15, +5 Standby	+5, ±12, +5 Standby, +12 Backup	+5
Megszakítási szintek száma	7	8	7	4	nincs
Vezérlőkijelölő szintek száma	4	1	5	1	32
Többprocesszor	igen	igen	igen	nem	igen
Hibajelek	AC FAIL, SYS FAIL, BUS ERROR	nincs	AC FAIL, SYS FAIL, BUS ERROR, PARITY	POWER OK	ECC
Tervezett bővítés	címmódosítók	nincs	címmódosítók	nincs	parancsmező
Önálló soros busz	igen	nincs	tervezés alatt	nincs	igen

SYSTEMS KLUB

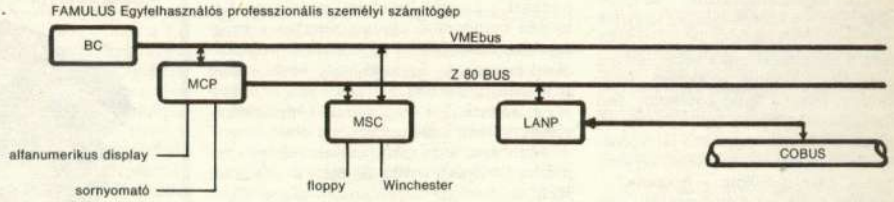
A VESTA kártyarendszer széles körben alkalmazható: professzionális személyi számítógép, fekete-fehér vagy színes grafikus terminál, egy- vagy többfelhasználós, többprocesszoros mikroszámítógép stb. összeállítására alkalmas. Mindezek az eszközök helyi adatátviteli hálózaton összekapcsolhatók (COBUS vagy ETHERNET), ütőképes, nagy hatékonyságú rendszerbe szervezhetők. (A COBUS az MTA SZTAKI által kifejlesztett 1 Mbit/s sebességű helyi adatátviteli hálózat, amely lehetővé teszi a számítástechnikai erőforrások egyesítését, egyes perifériák optimális kihasználását.)

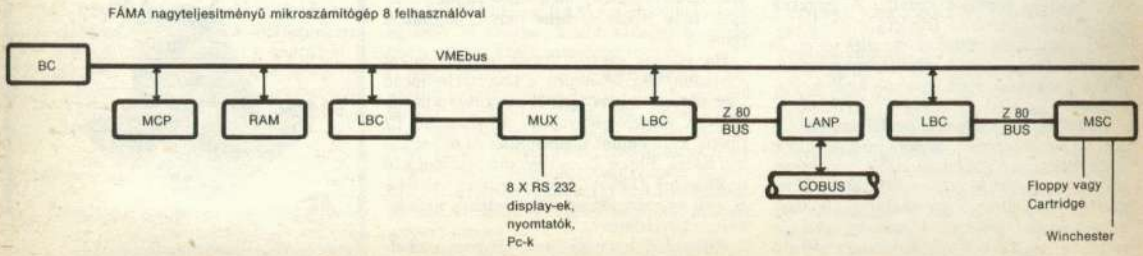
A VESTA KÁRTYARENDSZER ELEMEI

VME MCP Z8000:	16 bites CPU (Master) saját 0,5 Mbájt RAM-mal, memóriakezelő egységgel, 2 soros interfésszel
VME MCP 8086:	16 bites CPU (Master) lebegőpontos és I/0 segédprocesszorral, 0,5 Mbájt RAM-mal, 2 soros interfésszel
VME MCP 68010:	16 bites CPU (Master) 0,5 Mbájt RAM-mal, memóriakezelő egységgel, 2 soros interfésszel
VME MCP Z80:	8 bites CPU (Master) 64 kbájt RAM-mal, memóriabővítő regiszterrel, 2 soros interfésszel
VME RAM:	0,5 Mbájt RAM, változtatható adatszélességgel (8, 16, 32 bit)
VME BC:	Rendszervezérlő, EPROM-mal (16 db JEDEC kompatibilis 2732-től 27256-ig)
VME LANP2:	Helyi adathálózati csatlakozó (ETHERNET) és front-end processzor
VME Z80 VERSO:	VME busz és Z80 busz hátlap 10 kártyahellyel
VME BITMOVER:	32 bites mikroprogramozott aritmetikai és bit-map processzor
VME Z80 LBC:	Intelligens periféria processzor (Slave), 64 kbájt RAM-mal, soros és párhuzamos interfészekkel
Z80 LANP1:	Helyi adathálózati csatoló (COBUS, 1 Mbit/s)
Z80 MSC:	Háttértároló vezérlő SASI buszhoz és floppy controller
Z80 GD1:	Raszteres grafikus képernyővezérlő (1024 × 800 látható, 2048 × 1024 programozható képpont)
Z80 GDEXT:	Raszteres grafikus képernyővezérlő színes raszteres kijelzőkhöz (1024 × 1024 × 4 látható pont, 16 egyidejű szín egy 4096-os palettán; R, G, B analóg kimenet)
Z80 TEXPRO:	Nagy felbontóképességű alfanumerikus karaktergenerátor és programozható cursor
Z80 GDVLT:	Raszteres grafikus képernyővezérlő színes raszteres kijelzőkhöz (1024 × 1024 × 8 látható pont, 256 egyidejű szín egy 2^{24}-es palettán; R, G, B analóg kimenet)
Z80 MUX:	8 soros vonal multiplexere és 2 párhuzamos BSI port DMA-val

További részletes információk megadásával készséggel áll az Önök rendelkezésére az MTA SZTAKI Marketing.
Telefon: 667-424

Tájékoztatásul két alkalmazási példát mutatunk be a sok lehetséges változat közül:





MÓDSZEREK, FOGÁSOK, TEVÉKENYSÉGEK, PÉLDÁK

Ebben a rovatban alkalmazási módszerekkel, fogásokkal, az alkalmazáshoz kapcsolódó tevékenységekkel és példákkal foglalkozunk. Nem korlátozódunk azokra a felhasználókra, akiknek részben vagy egészben ez a foglalkozásuk.

Úgy véljük, hogy a komoly alkalmazások mindenkit érdekelnek. Ahogy végzetes pedagógiai tévedés az az általánosan elfogadott nézet, hogy a gyereket a mese érdekli, és a felnőttek elnagyolt, gyerekes firkáit élvezi a komolyan, pontosan kimunkált, részleteiben is megrajzolt, tanulságos grafika helyett, ugyanilyen tévedés a mi területünkön az, hogy a gyereket csak a játékprogramok érdeklik. Persze, ha csak azokhoz juthat hozzá. A kutya szereti a csontot. Kénytelen, hiszen a színhúst nem ő kapja.

Véleményünk szerint a gyereket valójában nem a mese, hanem az igazi világ érdekelné, ám ezt nem, vagy csak korlátozottan kapja meg. Nem a játékprogramokkal, hanem a valódi alkalmazásokkal foglalkozna, csakhogy nem jut hozzájuk.

Azt hisszük, hogy minden egészséges gyereket sokkal jobban érdekli az igazi, mint a játékvonat. Ha szabadon kérhetne ajándékot, minden, még el nem rontott gyerek igazi vonatot kérne, igazi űrhajót, igazi babakocsit, igazi babával.

A mi területünkön a hamis előítéleteken és begyökeresedett rossz szokásokon kívül semmi akadálya nincs annak, hogy az igazi kívánságokat valódi dolgokkal elégítsük ki. Így tehát rovatunk a gyerekek és a hivatásos szakemberek szolgálatát egyformán feladatának tekinti. Az első időkben fogalmazásmódban – de témaválasztásban nem! – a kezdők igényeit nagyobb mértékben fogjuk érvényesíteni.

Rovatunk a számítástechnika békés célokra való felhasználását kívánja terjeszteni. Megdöbbentő tény, hogy a fiatal korosztály első számítástechnika-alkalmazási tapasztalatait általában olyan, többségében primitív játékprogramokkal szerzi, amelyekben a másik lelövése, felrobbantása, azaz megölése – bocsánat! – megsemmisítése révén jut sikerélményhez. Ne felejtsük el, e programokat nem gyerekek, hanem infantilis felnőttek készítik. A gyerekek legtöbbször azért lesznek felnőtt korukra infantilisek, mert ilyen felnőttek nevelték őket, és azért nem fogják tudni a számítástechnikát önállóan alkalmazni, mert nem volt kitől megtanulniuk.

Reméljük, hogy ennek a csodálatos eszköznek, a számítógépnek – amely elsősorban nem játék, hanem az emberiség egyik legnagyszerűbb munkaeszköze – komoly alkalmazásai minden olvasónkat, a gyerekeket is „felnőtt" sikerélményhez juttatják. Hiszen az igazi „jó mulatság" csak a komoly „férfimunka" lehet.

A TÁRTAKARÉKOSSÁGRÓL

Természetes, hogy minél kisebb egy számológép tárterülete, annál könnyebb azt kinőni. Így a hajdan nagyon kínos tárgazdálkodási problémák, amelyek az utóbbi évtizedben lényegesen kevésbé voltak kellemetlenek, napjainkban a személyi számológépek (itt most kifejezőbb volna „népszámológépet" mondani) tömeges elterjedésével újból jelentkeznek.

Mit csináljunk velük? Menjünk bele ezekbe a gyakran idegölő, idő- és energiapusztító, nehéz feladványokba, vagy kerüljük el őket? Bármi áron hozzuk ki a maximumot gépünkből, vagy legyünk kényelmesek? Gyűrjük össze magunkat, zsugorodjunk kis takarónk alá, vagy nyújtózzunk ki kényelmesen, és gondoskodjunk nagyobb takaróról?

Ez nemcsak ízlés, nemcsak felfogás dolga. Nekünk, alkalmazóknak gondos, körültekintő elemzést kell végeznünk. A gyári szoftverfejlesztőtől elvárható, hogy bánjon takarékosan a hellyel. Ha van szabad helye, a nem „profi" programozó viszont legyen célratörő, és ne ugorjon be a feladványmániás, öncélú és egyoldalú takarékosságnak. Sohase tévessze szem elől: minden programnak elsődleges célja, hogy helyes eredményt lehessen vele létrehozni, nem pedig az, hogy minél kisebb helyen elférjen.

Nehezebb a kérdés akkor, amikor van helyünk, de nem annyi, hogy kényelmesen elférjünk. Menjünk nagyobb gépre, vagy fogjunk hozzá a tömörítés nehéz és gyakran hosszadalmas munkájához?

Ha van rá lehetőség, sokszor a nagyobb gép választása a jobb megoldás, mert nemcsak a tárral, hanem a költségekkel és saját időnkkel is takarékoskodni kell. Ha nincs elfogadható áron és időn belül elérhető nagyobb gép, akkor még egy szempontot mindenképpen meg kell vizsgálnunk: hányszor és hogyan kell majd gépünkkel a kitűzött feladatot megoldani. Kényelmi szempontok ugyanis nemcsak a programkészítésnél, hanem a program felhasználásánál is vannak. Egyszeri vagy kevés számú felhasználás esetében nyugodtan megterhelhetjük a „futtatási" munkát kisebb kényelmetlenségekkel. Nem kell megijednünk olyan megoldásoktól sem, hogy több (kényelmesen programozható) lépésben oldjuk meg a feladatot. Kezdők tévedése, hogy mindent géppel akarnak megoldani, mégpedig egyetlen programmal. Sokszor bizony hamarabb célhoz juthatunk több lépésben, néha papírt, ceruzát is igénybe véve.

Ha viszont sokszor fogják programunkat használni, elkerülhetetlen a részletekbe menő „aprómunka". Ilyet viszont semmiképp ne végezzünk határidő-fenyegetés árnyékában. Ez tipikus kutatómunka, elmélyedés és nyugalom kell hozzá. Ebben az esetben több gondot kell fordítanunk a felhasználás kényelmessé tételére is, ami egyszeri felhasználás esetében nem lényeges követelmény.

Befejezésül, hogy mindenki helyesen érzékelje a szóban forgó kérdés komolyságát és nehézségét, felhívjuk a figyelmet egy fontos – de megoldatlan – problémakörre és egy vele kapcsolatos, nagyon hasznos, gyakorlati tapasztalatszerzési lehetőségre.

Minden programnak vannak jellemzői, például tárigény, az utasítások darabszáma fajtánként, futási idő stb. Adjuk meg egy-egy jellemző extrémumait adott célra szolgáló, olyan programok halmazát tekintve, amely programok többi jellemzőjére valamilyen kikötés teljesül (például adott intervallumba esnek).

A fordított jellegű kérdés is nagyon fontos és érdekes. Melyek (mire használhatók) azok a programok, és mik ezek jellemzői, amelyeknek egyes jellemzőire adott előírásoknak kell teljesülniük?

Konkrétan például: mire használhatók azok a programok, amelyeknek tárigénye rendre n, n+1, ..., n+k-1 bájt. Vagy például, mire használhatók azok a programok, amelyek rendre pontosan (legfeljebb stb.) k_1, k_2, ..., k_n darab m_1, m_2, ..., m_n műveletet használnak.

Felhívjuk a figyelmet arra, hogy e nehéz problémák – noha a gyakorlatban nagyon hasznos volna, ha ismernénk megoldásukat – az egyszerűbbek közé tartoznak. Ne felejtsük el, hogy lényegében minden programjellemző változhat is. A futási idő függhet a bemenő adatok értékétől. Ugyanígy a tárigény is. Sőt a tárigény a futás alatt is változhat stb. Így tehát nemcsak jellemzőértékek, hanem ezek eloszlásfüggvénye is szerepet kap e problémákban, melyekkel való megbarátkozáshoz és egyikének másikának megoldásához sok sikert kívánunk.

POGÁNY CSABA

Kabalapályázatunkra érkezett

5. *Török József, Pápa*

A TZ 80 személyi számítógép

A TZ 80 típusú professzionális személyi számítógépből a fejlesztő és gyártó, a Műszertechnika GM már több, mint 60 darabot készített, és előrehaladott tárgyalásokat folytat a gép exportálásáról is.

Az első benyomások

Amikor az 1983. évi tavaszi BNV-n először találkoztam a TZ 80-nal, azt gondoltam: íme itt van egy újabb, amellé a Magyarországon kifejlesztett, több mint 50 fajta mikroszámítógép mellé, amelyeket fáradságos munkával megtervezték, majd néhány készülék legyártása után a terveket fiókba tették, mert megfelelő eszközök híján ipari sorozatgyártásba soha nem kerülhettek. A gépet tervező és gyártó Műszertechnika GMK tagjai azonban lelkesedésükkel és munkabírásukkal megcáfolták gondolataimat: Magyarországon ma már több mint 60 berendezés üzemel.

1983 őszén a SZÁMOK (a SZÁMALK Oktatási Irodája) a számítógép tesztelésére, vizsgálatára kölcsön kapott egy TZ 80-at. Az alábbiakban a próbaüzem alatt szerzett tapasztalatokat felhasználva ismertetem a Műszertechnika által gyártott berendezést.

A TZ 80-ról általában

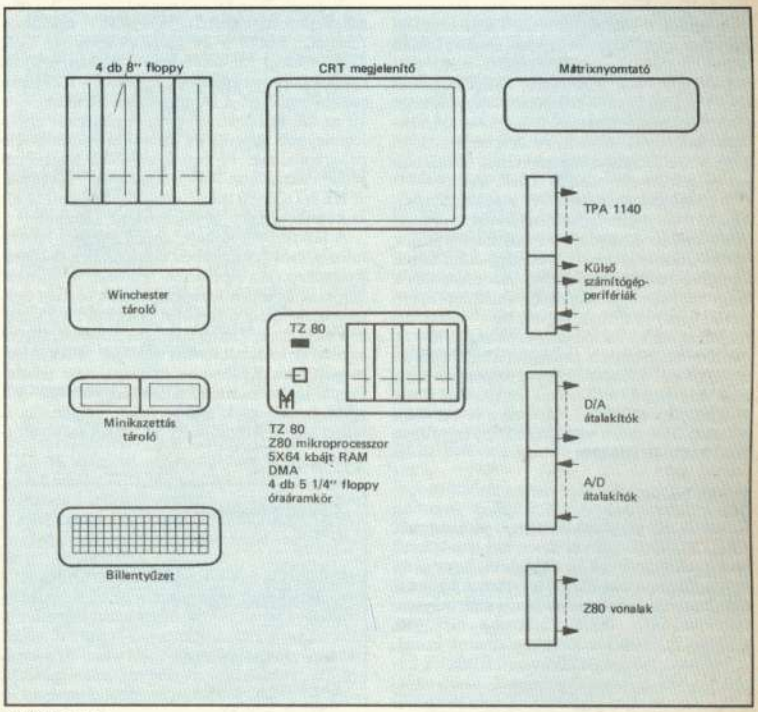
A TZ 80 a 8 bites személyi számítógépeknek azon kategóriájába tartozik, amelyek háttértárolóként egynél több hajlékony mágneslemeztárolót alkalmaznak, és amelyeken több különböző fajta, a gép működését felügyelő operációs rendszer futtatható. Az előzőekből következik, hogy a gépet több, az adott feladatmegoldáshoz legjobban alkalmazható, magas szintű programnyelven lehet programozni.

Belső felépítése lehetőséget ad egy, a világ számítógéppiacán széles körben elterjedt mikroszámítógépen, a TRS 80-on megírt programok futtatására és alkalmazására, így a gépen használható programok jelentős része eleve biztosított. A hazai piacot figyelembe véve szerencsésnek mondható az a körülmény, hogy a TZ 80 BASIC szinten felülről kompatibilis a HTSZ által gyártott HT–1080Z iskolaszámítógéppel; így azok, akik ezt a gépet ismerték, minden különösebb nehézség nélkül programozhatják a TZ 80-at is. Majdnem ugyanez a helyzet a PRIMO elnevezésű géppel is. Azok a felhasználók, akik ezeknek a gépeknek a lehetőségeit már kinőtték, programjaik nagy részét változtatás nélkül át tudják vinni a jóval több szolgáltatást nyújtó TZ 80-ra.

A gép alkatrészeinek jelentős hányada szocialista gyártmányú: például a mikroprocesszort az NDK-ban készítették, a képcsövet a TUNGSRAM, a mágneslemezegységet a MOM, a billentyűzetet a TÁKI, a géphez csatlakozó nyomtatókat a Telefongyár, illetve a DATACOOP szövetkezet gyártja.

A központi egység

A formatervezett dobozban elhelyezkedő központi egység a CPU, memória- és perifériavezérlő nyomtatott áramköri kártyákat, a tápegységet és a hajlékony mágneslemezegységeket tartalmazza *(1. ábra)*.



A TZ hardver

A *CPU kártya* NDK gyártmányú U880 típusú (Z80-nak megfelelő) mikroprocesszorral működik. A 2 MHz-es központi áramkörön kívül vonalmeghajtó és memória-áramköröket tartalmaz. A memóriák elsősorban kártyaelcseréléshez és javításhoz szükségesek. Ez a tárolhelyezés a tervezők előrelátását bizonyítja, mert a CPU kártyákra általában nem szoktak memóriát beültetni, így azok bemérése csak nehézkesen valósítható meg. További lehetőség, hogy a CPU kártyára 64 kbájtnyi RAM áramkört helyeznek el, ezért 64 kbájtnál kisebb tárkiépítés esetén nincs szükség külön memóriakártyára.

Nagy félvezetőmemória-igény esetén a különálló *memóriakártya* 4 × 64 kbájt adatot tud tárolni, maximális kiépítésben tehát 5 × 64, azaz 320 kbájt lehet a gép kapacitása. A központi egység kártyák külön szolgáltatása, hogy 12 kbájt EPROM-ban tartalmazhatnak BASIC programnyelvű fordítót is, így egyszerű felhasználások, tanulóprogramok esetén nincs szükség mágneslemezre.

A *perifériakártyák* a géphez csatolható kiegészítő berendezéseket (mágneslemezegység, nyomtató, kijelző stb.) vezérlik. Itt helyezkedik el egy óraáramkör, amely a pontos idő megjelenítését teszi lehetővé. A számítógép működéséhez szükséges egyenfeszültségeket kapcsolóüzemű tápegység állítja elő.

A hajlékony mágneslemezegység

A TZ 80 alapkiépítésben 2 darab normál szélességű, 5 ¼"-os mágneslemezegységet (például a MOM gyártmányú MF 900/1800 típus) tartalmaz. Ezek kapacitása összesen 360 kbájt. Lehetőség van arra, hogy 4 darab félszélességű mágneslemezegységet építsünk a központi egységbe, sőt arra is, hogy ezek dupla írássűrűségűek és dupla sávsűrűségűek legyenek, így a háttérkapacitás 1,5 Mbájtra növelhető.

Bizonyos esetekben még ez a 1,5 Mbájt adat is kevésnek bizonyulhat. Ekkor lehetőség van arra, hogy 10 Mbájtos Winchester-tárolót, tehát nagy kapacitású, rögzítettlemezes háttértárat kapcsoljunk a berendezéshez. Ez azért is előnyös tulajdonság, mert a MOM már lépéseket tett ilyen tárolók kifejlesztésére, így ezeket

hamarosan hazai piacról is be lehet majd szerezni. Ha a felhasználó standard 8" mágneslemezzel rendelkező régi számítógépét kívánja TZ 80-nal helyettesíteni, akkor annak sincs akadálya, hogy standard 8" lemezegységeket csatoljon a TZ 80-hoz.

Konzol perifériák

A bemeneti konzol periféria – ami az információkat a számítógép központi részébe továbbítja – a széles körű elterjedtsége miatt már „szabványos"-nak mondható, hallgenerátoros TÁKI gyártmányú billentyűzet. Nagy tömegű numerikus adat feldolgozásához külön számbeviteli billentyűk állnak rendelkezésre. Kár, hogy a funkcióbillentyűk nincsenek kihasználva, így feleslegesen díszítik a klaviatúrát.

A kimeneti konzol periféria a Műszertechnika GMK által kifejlesztett, tetszetős formatervezett házban elhelyezett, 38 centiméter átlójú, hosszú utánvilágítással rendelkező, zöld színű, katódsugárcsöves megjelenítő. Az információk (nagy- és kisbetűk, számok, valamint egyéb ASCII jelek) 16 sorban, soronként 24 karakterrel jelezhetők ki. Jó felbontóképességű: 48 × 48 mátrixpont méretben szemigrafikus ábrázolás is lehetséges. Előnyös, hogy a képernyő fényerejét az előlapon lévő csavar segítségével állítani lehet, így a kezelő saját maga is be tudja állítani a környezeti megvilágításhoz legjobban alkalmazkodó fényerőt.

Egyéb perifériák

A TZ 80 alkalmas minden párhuzamos, CENTRONICS típusú illesztéssel rendelkező nyomtató vezérlésére. Ez azt jelenti, hogy nemcsak a Telefongyár TMT 12× típusú nyomtatója, hanem a legismertebb japán mátrixnyomtatók (EPSON RX80, Seikosha GP 100, C.ITOH 8510 stb.) is csatlakoztathatók hozzá.

Terepen, szélsőséges környezeti feltételek között végzett mérésadatgyűjtésnél, ahol hajlékony mágneslemez alkalmazása nem biztonságos, adattárolásra Philips gyártmányú minikazettás adatrögzítő is felhasználható, mivel ez is szerepel a TZ 80 perifériaválasztékában.

Több TZ 80, illetve nagyobb számítógépek között szabványos RS 232A, illetve V24 vonalakon kapcsolat építhető ki. Egy már megvalósított alkalmazásban a berendezés TPA 1140 típusú számítógép intelligens terminájaként üzemel.

Ipari folyamatok irányításához a géphez analóg-digitál, illetve digitál-analóg jelátalakítók kapcsolhatók. Miután a mikroprocesszor jeleit megfelelő nagyságú áramot biztosító, ún. meghajtó áramkörök közbeiktatásával egy külső csatlakozóra kivezették, a TZ 80-hoz egyszerűen illeszthető tetszőleges típusú számítógép-periféria vagy egyéb berendezés.

Az operációs rendszerek

A TZ 80-on többféle, a számítógépek működését felügyelő programrendszer, ún. operációs rendszer futtatható. Ezek közül a legismertebbek a TRSDOS, a NEWDOS/80, a CP/M és az ezzel egyenértékű MSYS programrendszerek. Az előzőeken kívül a TZ 80 rendelkezik még több feladat egyidejű, de különböző prioritásszintű, illetve több számítógép összekapcsolását lehetővé tevő multitask, illetve multiprocessing operációs rendszerekkel, valamint PASCAL nyelvű és folyamatokból érkező jeleket feldolgozó felügyelőprogram-rendszerekkel is.

A NEWDOS/80

A NEWDOS/80 egyfelhasználós, 28 kbájt mágneslemez-területet elfoglaló operációs rendszer. Ebből a 28 kbájtból azonban csak közel 4 kbájt van állandóan a tárban, a többi csak akkor olvasódik be a lemezről – 1,28 kbájtos egységekben –, ha működésére szükség van. Ez az ún. átfedéses (overlay) technika a lehető legnagyobb tárterületet biztosítja a felhasználó programjainak. A kezelő számára legérdekesebb parancsok a táblázaton láthatók. A NEWDOS/80 tulajdonságait, szolgáltatásait és kezelhetőségét jól jellemzik az alábbiak.

A fájlok 5 × 256 bájt, azaz 1280 bájt hosszú információs egységekben tárolódnak a mágneslemezeken. Az operációs rendszer a fájlokat nemcsak egyetlen lemezen, hanem sorban egymás után minden forgatóegységben lévő lemezen megkeresi. Fájlok utólagos növelése, illetve az általuk lefoglalt terület utólagos csökkentése megtörténhető. Különösen nyilvántartási feladatoknál jól használható a „Módosított fájl" jelzőbit, amely csak akkor billen „1"-be, ha a fájlba adatot írtunk, vagy ha adatot módosítottunk, és így a napi fájlkarbantartásnál elegendő csak a „Módosított fájl"-okat újra archiválni. A fájlok logikai rekordhossza 1 bájt és 4 kbájt között változhat.

ATZ80 szoftver

Operációs rendszerek
CP/M
MSYS
TRSDOS
NEWDOS
Több feladatos OS
PASCAL OS
Valósidejű OS
Több számítógépes OS

Kiegészítő programok
Rendszergeneráló
Rendszermásoló
Fájl-, lemezmásoló
Szövegszerkesztő
Z80 ASSEMBLER/DISASSEMBLER
8080 ASSEMBLER/DISASSEMBLER
DEBUGGER
Adatkezelő
Táblázat/képernyőkép kezelő

Programnyelvek
BASIC interpreter (ROM)
BASIC interpreter (lemezes)
BASIC compiler
PASCAL
COBOL
FORTRAN
C-nyelv
TZ 80

Felhasználói programok
Készletgazdálkodás
Főkönyvi könyvelés
Rendelés-nyilvántartás
Munkaidő-nyilvántartás
Bérszámfejtés
Személyi nyilvántartás
Matematika, statisztika
Játék

APPEND:	Fájlok egymáshoz kapcsolása
ATTRIB:	Fájltulajdonságok definiálása
AUTO:	Bekapcsolás után azonnal meginduló program definiálása
BLINK:	Kurzor villogás be/kikapcsolás
CLEAR:	Memóriatörlés
CLOCK:	Másodpercenkénti időkijelzés be/kikapcsolása
CLS:	Képernyőtörlés
COPY:	Fájl-, lemezmásolás
CREATE:	Fájlhelyfoglalás
DATE:	Dátummegjelenítés
DEBUG:	Hibakezelő üzemmód be/kikapcsolása
DIR:	Fájlnyilvántartási jegyzék megjelenítés
DUMP:	Memóriatartalom kivitele lemezre
FORMAT:	Mágneslemez-formattálás
FREE:	Felhasználható lemezterület megjelenítés
HIMEM:	Legnagyobb memória definiálás
JKL:	Képernyőkép-nyomtatás
KILL:	Fájltörlés
LIB:	Rendszerparancsok megjelenítése
LIS:	Fájllistázás
LOAD:	Fájlbetöltés a memóriába
PAUSE:	Programfutás megállítása
PRINT:	Fájlnyomtatás
PROT:	Fájlvédelem-beállítás
PURGE:	Több fájl törlése
RENAME:	Fájlátnevezés
ROUTE:	Periféria-üzembeállítás
STMT:	Képernyő üzenetkijelzés
TIME:	Időmegjelenítés

A NEWDOS/80 legfontosabb parancsai

A fájlok a szokásos DIR/SYS és írható-olvasható/csak olvasható attributumokon kívül hatékony védelemmel rendelkeznek. Létrehozáskor minden fájlhoz hozzárendelhető egy kódszó, és minden további hozzáférés csak e kódszó megadásával hajtható végre. Sőt hozzáférési szintek is definiálhatók, így létezik csak futtatható, csak olvasható, írható/olvasható, átnevezhető, törölhető stb. fájl is. Ezenkívül minden mágneslemezt el lehet látni kódszóval. Ekkor bizonyos rendszerparancsokat (például lemezmásolás) csak e lemezkódszó megadásával hajt végre a rendszer. A fájlnyilvántartási jegyzék megadhatja nemcsak a fájl nevét, hanem összes jellemzőjét, beleértve a helyét a lemezen, a lefoglalt, a valóságban elfoglalt lemezterület nagyságát, a fájlvég pontos helyét stb.

Kényelmessé teszi a rendszerkezelést a LIB, a DATE, a CLOCK és a TIME parancs. Ez utóbbi kettő között az a különbség, hogy a TIME csak egyszer írja ki, a CLOCK másodpercenként jeleníti meg a pontos időt. Az AUTO parancs előre megadott művelet automatikus végrehajtását teszi lehetővé. Ha például a számítógépet csak egy BASIC nyelven megírt, nyilvántartási feladatot ellátó program futtatására használjuk, AUTO BASIC RUN „NYILVÁNTARTÁS" paranccsal beállítható,

hogy feszültségbekapcsolás után azonnal a nyilvántartási program futása induljon és ne az operációs rendszer bejelentkezésével kezdődjön a működés.

Az operációs rendszer másolóparancsa, a COPY, mind fájl-, mind lemezmásolást végre tud hajtani, akár logikai fájlonként, akár fizikai szektoronként. Ha a céllemez formattálatlan, a másolás mellé formattálás is kérhető.

Ugyancsak hasznos szolgáltatás az egyetlen lemezegység segítségével történő másolás, amikor is az operációs rendszer maga kéri a másolandó lemezeket. A COPY parancshoz hozzárendelhető olyan fájl, amely a másolandó fájlok neveit, de olyan is, amely a nem másolandó fájlok neveit tartalmazza. A fenti néhány példa is bizonyítja, hogy a NEWDOS/80 jól kezelhető, a szokásos mikrogépes operációs rendszereknél jóval több paranccsal rendelkező felügyelőprogram.

A NEWDOS/80 kiegészítő programjai

A NEWDOS/80-hoz csatlakozó kiegészítő programok: az EDITOR/ASSEMBLER, a DISASSEMBLER, a LOADER, a rövid szövegek előállítására alkalmas CHAINBUILD, a lemeznyilvántartás vizsgálatára alkalmas DIRCHECK és a futás alatti nyomtatást biztosító ASPOOL. A DEBUG program lehetővé teszi, hogy egyidőben 3 × 64 bájt memóriamező és a mikroprocesszor összes regiszterének tartalma kijelződjön a képernyőn. Lépésenkénti programfuttatásnál a jól működő szubrutinok végrehajtása egyetlen programlépésnek értelmezhető. Lehetőség van mind a memória, mind a lemeztartalom bájtonkénti módosítására.

Kár, hogy a NEWDOS/80 nem rendelkezik az előző parancs megjelenítésének képességével, mert akkor az előzőleg hibásan beadott hosszú parancssorok javítása sokkal kényelmesebb lenne. A futtatható programnyelvek: BASIC interpreter, BASIC compiler, PASCAL, COBOL, FORTRAN, C-nyelv.

BASIC interpreter

A legtöbb felhasználó BASIC nyelven programoz, ezért érdemes néhány szót ejteni a TZ 80 BASIC-jéről. A TZ 80 két, egymással kompatibilis BASIC interpreterrel rendelkezik. Az egyik 12 kbájtot foglal el és ROM memóriába helyezhető. Egyszerű alkalmazásokra, tanulásra, játékra ez a fordítóprogram éppen megfelelő.

Nagyobb lélegzetű, bonyolultabb feladatok megoldására a mágneslemezen elhelyezkedő „Továbbfejlesztett BASIC" interpreter használatos, ami felülről kompatibilis a ROM–BASIC-kel. Ez mágneslemezen 16 kbájt, a memóriában 8 kbájt területet foglal el és a NEWDOS/80-nal hasonló átfedéses (overlay) programfuttatási módszerrel működik. A „Továbbfejlesztett BASIC" a nagyszámítógépek BASIC-jeinek majdnem minden jó tulajdonságával rendelkezik. Például lehetőséget nyújt minden operációs rendszer parancs BASIC programból való kiadására. Tetszőleges azonosító, változó, parancs vagy utasítás programban elfoglalt helye SEARCH paranccsal megkereshető. Hét különböző típusú fájlszerkezetet (soros/tetszőleges elhelyezkedésű, rögzített/tetszőleges, előre/hozzáféréskor definiálható rekordhosszúságú stb.) tud kezelni, így az alkalmazáshoz legjobban illeszkedő típus mindig kiválasztható.

CP/M kompatibilis operációs rendszer

A CP/M a leginkább elterjedt mikroszámítógépes operációs rendszer. A TZ 80-on futtatható egy, a CP/M-mel azonos tulajdonságokkal rendelkező, MSYS nevű (MIKROPO) rendszer is. Ennek szolgáltatásai a szokásos CP/M beépített parancsokon kívül a következők (a zárójelben lévő elnevezések az eredeti CP/M, illetve az MSYS jelölésekre utalnak):

- Rendszergeneráló (MOVCPM, MLOAD)
- Rendszermásoló (SYSGEM, MGEN)
- Fájlmásoló (PIP, MPIP)
- Szövegszerkesztő (ED, MEDIT)
- Rendszerállapot-lekérdező és módosító (STAT, MSTAT)
- 8080 ASSEMBLER (ASM, ASM80)
- 8080 DEBUGGER (DDT, MDT)
- Z80 ASSEMBLER (–, ASMZ 80)
- Z80 DEBUGGER (–, MDZ)

Az MSYS lehetővé teszi, hogy minden CP/M alatt futó magas szintű programnyelv (BASIC, PASCAL, FORTRAN, COBOL, C-nyelv stb.) a TZ 80-on is futtatható.

Felhasználói programok

A TZ 80-on használható programok köre rendkívül széles. A gép hardver szempontból a TRS 80 személyi számítógéppel kompatibilis, így az összes TRS 80-ra kifejlesztett programot értelmezni tudja. A szoftver oldalról biztosított CP/M kompatibilitás viszont az összes CP/M-re megírt program futtathatóságát teszi lehetővé.

A gyártó ezenkívül saját maga is kifejlesztett néhány, a hazai viszonyokat figyelembe vevő alkalmazói programot. Néhány tipikus felhasználás: főkönyvi könyvelés, készletfigyelés, rendelés-nyilvántartás, munkaidő-nyilvántartás, bérszámfejtés, személyi nyilvántartás, szöveg- és levélszerkesztés, táblázatszerkesztés, adatbázis-kezelés.

A gép ezeken kívül még számos játékprogrammal, matematikai, statisztikai, ügyviteli programokkal is rendelkezik.

Üzemeltetési tapasztalatok

A berendezés üzembiztonsága nagyon jó. A vizsgálati időszak alatt semmilyen meghibásodás nem történt, pedig a gépet kifejezetten tartós üzemű „nyúzópróbának" vetettük alá. A készülék bekapcsolása kulcsos kapcsolóval történik. Ellentétben más gépekkel, itt a kulcs nemcsak kikapcsolt, hanem bekapcsolt üzemállapotban is kihúzható, így hosszú futási idejű programok működése alatt a TZ 80 a program külső behatásra történő megszakításának veszélye nélkül magára hagyható.

A klaviatúra gombjainak kezelése kényelmes, de a kihasználatlan funkcióbillentyűk zavarólag hatnak. A központi egység háza az általunk kipróbált példánynál elég terjedelmes és kevésbé tetszetős volt. Újabban a gépeket már asztalfiókba szerelve is szállítják.

A géppel együtt szállított magyar nyelvű dokumentáció részletes és könnyen érthető. A különlegesen fontos műveletek (lemezformattálás, rendszerlemez-másolás stb.) külön fejezetben a gépkönyv elején találhatók. A katódsugárcsöves kijelző a normális háztartási tévékészülékekhez hasonlóan nem kellemes, vagy nagyfrekvenciás sípoló hangot adott. (A gyártótól kapott tájékoztatás szerint a konstrukció módosításával ezt a jelenséget megszüntették. – Szerk. megj.)

A berendezés ellentétben a hazai gyakorlattal, ahol a hardver mellé jó néhány felhasználói programot is „kötelező" megvenni, a TZ 80 részegységenként is megvásárolható, így a hozzá kapcsolt perifériák, illetve a tárkapacitás az igények növekedésével folyamatosan bővíthető. Kár, hogy a gép hibaüzenetei angol és nem magyar nyelvűek. (Sajnos a hazai számítástechnikai gyakorlatban az angol nyelvű operációs rendszer üzenetek az elterjedtek. – Szerk. megj.)

A géphez kapható programok köre a TRS 80, illetve a CP/M kompatibilitás miatt rendkívül széles. Előnyös az is, hogy azok a középiskolából kikerült tanulók, akik az iskolaszámítógépen tanultak meg programozni, tapasztalatukat és saját kifejlesztett programjaikat a TZ 80-on használni tudják. A gép operációs rendszereinek szolgáltatásai jobbak, mint az azonos kategóriájú más gépeké. Egy, a számítástechnika alkalmazását most kezdő felhasználó számára talán túl bő a gép által nyújtott szolgáltatásválaszték. Haladó programozó munkáját azonban nagymértékben tudják segíteni a TZ 80-ba beépített funkciók és egyéb szolgáltatások.

A gyártótól kapott tájékoztatás szerint folyamatosan bővítik gyártmányválasztékukat. Hordozható kivitelű, 8 bites gépük – Magyarországon elsőként – már megjelent, a 16, illetve 32 bites számítógépük pedig hamarosan piacra kerül. (A hordozható változatot valamelyik következő számunkban ismertetjük. – Szerk. megj.)

SZŰCS TAMÁS

ADOK – VESZEK – CSERÉLEK

Ebben a rovatban rövid, szöveges, a mikroszámítógépekkel kapcsolatos hirdetéseket közlünk. A díjszabás: közületeknek gépelt soronként (60 karakter) 100,– Ft, magánszemélyeknek az első sor 50,– Ft, minden további sor 20,– Ft. Az NJSZT tagjainak az első három sor ingyenes. Hirdetéseiket a szerkesztőség címére várjuk.

- **SINCLAIR SPECTRUM** 8 kbájt géphez szuper játékprogramok, képernyőkezelő programok, textprogramok, nyelvprogramok (Beta-Basic, Snail-LOGO, Forth, Pascal, Aspect 4.2, Assembler, Disassembler, Spec Bug2–48k), valamint különböző grafikai programok, beszéd-digitalizáló, másoló programok kaphatók. Kérésre tájékoztatót küld: Szatai László, Atkár, Fő u. 58. 3213.
- **TEXAS INSTRUMENTS 99/4A**-val rendelkező klubtársak jelentkezését várjuk „Cartridge-csere" jeligére a szerkesztőségbe.
- **VC–20 TANKCSATA** (Panzerschlacht) játékprogram gyári kazettán eladó. Szathmári Tamás, Miskolc, Próbatétel 11. 3508.

A Számítástechnika című lap decemberi száma közölte az Ipari Minisztérium Innovációs Alapja, a KSH Számítástechnika-alkalmazási főosztálya és a MIGÉRT pályázatát, részletesen ismertetve a szövegfeldolgozó rendszerre vonatkozó kiírás feltételeit. A helyes magyar nyelv és a szakmai terminológia, ékezetes-ékes betűink-hangzóink és az őket csonkító bemeneti/kimeneti eszközök szembekerültek egymással az egyébként örvendetes fejlődés évtizedei során. Valamit tenni kell. Ezt tükrözik – sőt már valamelyes reményeket is kelthetnek – az itt következő írások, amelyek nevetésre és bosszankodásra ingerelnek.
Reméljük, hogy egyéb cselekvésre is; olvasóink közül kit-kit a maga hatáskörében.

Kornyezetunk vedelmerol

Igen, azaz szellemi környezetünk védelméről – vagy inkább védtelenségéről – szeretnék itt szólni. Az utolsó adalék, amely írásra ösztönöz, az a cédula, amelyet a Neumann János Számítógéptudományi Társaság pár napja érkezett tagdíjbefizetési csekkjére ragasztottak. Ennek szövegét ékezetes betűket nem tartalmazó számítástechnikai eszközzel állították elő. Bárki mondhatná erre: mit csodálkozik, hol él maga? Hiszen az országban használatos számítástechnikai eszközök zöme ilyen. Kis hányadukon meg lehet ugyan találni az ékezetes betűk egy részét, de a teljes magyar betűkészlettel rendelkező eszköz olyan ritka, mint a fehér holló.

Nem csodálkozom, de azért megkérdezem: tényleg, hol élek én? Egy olyan országban, ahol a nyelvművelés közügy; ahol a szaknyelv, a szakkifejezések magyarításán is tömegek szorgoskodnak; ahol a számítástechnika társadalmi hatásait mélyenszántó írások és beszédek taglalják; ahol évtizede van Számítástechnikai Központi Fejlesztési Program; ahol – ugyanakkor – lassan elfelejtik vagy meg sem tanulják az emberek a hosszú magánhangzók használatát, egyebek között azért, mert szinte mindenki és egyre gyakrabban találkozik a hiányos betűkészletű gépelt szövegek mellett a hiányos betűkészletű számítástechnikai berendezéseken készült szövegekkel.

Ezt a szellemi környezetszennyezést folytatja – kisebb vagy nagyobb körben – szinte valamennyi, szöveges szolgáltatást nyújtó számítógép-alkalmazó. „Hatósugara" miatt említést érdemel az OTP, a Posta, a Fővárosi Tanács (közüzemi számlák), a számítógépes könyvtári rendszerek zöme, a Televízió (képújság).

A számítástechnika tömegméretű, országos elterjedésével mind többször kényszerülünk ilyen torz írások olvasására, ami egyrészt növeli a nyelvrombolás veszélyét, másrészt – indokoltan – megutáltatja a számítástechnikát a szép magyar nyelvért síkraszálló laikus felhasználókkal. (Remélhetőleg mindez zavarja azokat a szakembereket is, akik úgy érzik, hogy a technikai eszközöket kell az emberek igényeihez alakítani, és nem fordítva.)

A kialakult helyzet megítélésem szerint abnormális: rákényszerítjük a gépvásárlókra a korcs íróműveket, a lakosságra pedig az ezeken írt dokumentumokat.

A váci vízszennyezés kormányzati szintű mulasztások és helyi felelőtlenségek együttesének következménye volt. A számítástechnika jelenlegi káros, nyelvrontó mellékhatása is hasonló okokra vezethető vissza. Azonban ennek boncolgatása helyett inkább a helyzet normalizálását szeretném sürgetni.

Utalok a Számítástechnika című folyóirat 1981. decemberi számában Papp Ferenc akadémikus aláírásával megjelent cikkre: Akadémiai állásfoglalás – magyar nyelvű szövegek számítógépes ábrázolásának egységesítéséről. Az ebben foglaltakat is figyelembe véve el kellene végre indítani a rendezési folyamatot!

A Neumann János Számítógéptudományi Társaság tagjai között tudhatja mindazokat a természetes és jogi személyeket, akik (amelyek) tenni tudnának ennek érdekében. A mi szakmánk okozza ezt a szellemi környezetszennyezést, szakmai problémáink megoldatlanságának terheit toljuk így át a tiltakozni képtelen társadalomra. Úgy vélem, hogy másra át nem hárítható és tovább nem halogatható kötelességünk a cselekvés.

Az NJSZT tagjaként szeretném, ha társaságunk – alapszabálya szellemében – élére állna ennek az anyanyelvünk védelmét szolgáló megmozdulásnak, és szakmai felkészültségével, szervező erejével hozzájárulna a munka eredményességéhez.

SIPKA LÁSZLÓ

Pont, pont,

„Hogy került a raktáramba 400 méter *fókabél*?" – kérdezte telefonon ingerülten egy főkönyvelő a Kecskeméti Számítástechnikai és Ügyvitelszervező Vállalattól néhány évvel ezelőtt, miután átnézte a számítógéppel készített, szöveges készletkimutatást.

A reklamáció jogos is volt, meg nem is, mert a raktárban nem volt fókabél, de volt 400 méter *fókábel*, amelyet a gépi kimutatás FOKABEL-nek tüntetett fel. Volt továbbá ALLOLAMPA (állólámpa), CSOALLVANY (csőállvány), BOR (bőr), FUZOGEP (fűzőgép), KEMÉNYITO (keményítő), LADA (láda), KOSZORU (köszörű), SOPRU (söprű) és VESO (véső) is. Ezeket a szavakat azonban a szakértő szem nem nézte el.

A telefon ma jogosabban szólhatna, mint eddig bármikor, holott az a gép már FÓKÁBEL-t, BÖR-t, CSÖÁLLVÁNY-t, KÖSZÖRÜ-t és VÉSÖ-t ír, mert a nagyszámítógépek sornyomtatóinak többségénél a legszükebb ékezetes betűhalmazt – az Á, É, Ö, Ü betűket – a kiírható matematikai vagy műveleti jelek rovására beépítették a jelkészletbe.

A ladából láda, a koszorúból köszörű, a szelesből széles, az uszoból üszö lett, megkülönböztethető egymástól a verés, a veres és a véres, de további gondot jelent, hogy szöveges adattárolásnál és kiiratásnál csak következtetéssel tudjuk eldönteni, hogy irt vagy írt, koros vagy kóros, joga vagy jóga, őrül vagy őrül, török vagy török vagy török, turán vagy túrán, üröm vagy ürröm-e, amiről szó van.

A kiragadott nyelvi példáknál bizonyára lehet értelemzavaróbbakat, mosolyogtatóbbakat is találni, de a probléma mindenképpen komoly. Napjainkban kétarcú jelenség figyelhető meg. Egyfelől nyelvészeink, pedagógusaink, a sajtó és a közigazgatás küzd nyelvünk helyes alkalmazásáért, tisztaságáért, rendelet szabályozza a magyar nyelv használatát a közokiratokban, szabványosították a Magyarországon forgalomba hozható teljes betűkészletű írógépek jelhalmazát és billentyűelrendezését. Másfelől a modern technika mind szélesebb elterjedése kapcsán egyre gyakrabban jelennek meg számítógépes adatfeldolgozások eredményeképpen született szöveges kimutatások, amelyekben nevek, földrajzi nevek és más szövegrészek szűkített ékezetes jelhalmazzal vagy teljesen ékezet nélkül szerepelnek. Horribile dictu, még postai címzések is. (Például Török Ödön, Szárszó, Köszörű u. 2. így: TOROK ODON, SZARSZO, KOSZORU U. 2.)

A probléma megoldására a hazai hardver- és szoftverfejlesztők eddig nem sok energiát fordítottak, pedig a környező latinbetűs szláv országokban és a skandináv népeknél is használnak speciális ékezetes jeleket (a cseh nyelvben például még a magyarnál is többet), és talán lennének hasznosítható tapasztalataik.

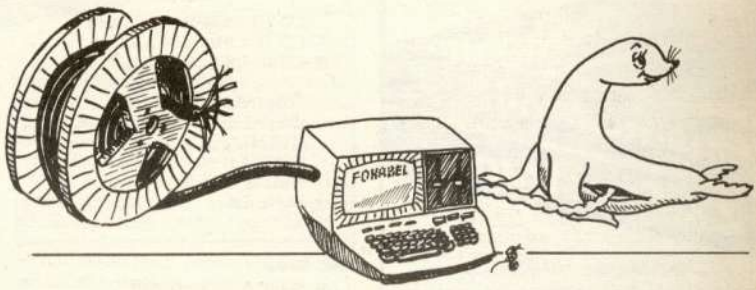
Elgondolkodtató, hogy nagy, nyugati gyártócégek (Siemens, Olivetti, IBM) egy-egy megvásárolt berendezésen is megoldják ezt a kérdést néhány hét leforgása alatt, ugyanakkor a kb. hatvanféle, Magyarországon konstruált „magyar" mikroszámítógép között alig van egy-kettő, amely körülményesen bár, de kezeli az ékezetes jelhalmazt (például a FLOPPY-

esszőcske...

MAT). Angolul mindegyik tud, sok németül is, és olyanok is akadnak, amelyek a teljes arab és/vagy cirill jelkészletet képesek produkálni, de nem tartalmazzák a legfontosabb ékezetes magyar karaktereket sem. Nyilvánvaló, hogy a fejlesztők nem hazai alkalmazásra szánták ezeket a berendezéseket, hanem az exportpiacot célozták meg velük. Még a VIDEOTON gyárnak sincs elfogadható eredménye ezen a téren.

A probléma felvetésének az ad különös aktualitást, hogy a megoldással a huszonnegyedik órába léptünk. Az elmúlt években üzembe helyezett számítógépek „termőre fordulásával" és a mikro-számítástechnika tömeges elterjedésével a műszaki, gazdasági, közigazgatási, társadalmi életbe, sőt a magánéletbe is széles áradatban tör be a számítástechnika-alkalmazás.

Amíg csak vállalatok, hivatalok használták a számítógépet, nagyobbára numerikus feladatok megoldására, és a berendezés összesen helyett ÖSSZESEN-t írt, besz. ár, elsz. ár (beszerzési, elszámoló) helyett BESZAR-t, ELSZAR-t, néhány nevet rosszul, de környezetében érthetően jelzett ki, ez belső ügy volt, és a gép haszna mellett ez a szépségháha eltörpült. Most azonban ezrével kerülnek a piacra nagy kapacitású mikroszámítógépek, és a szóban forgó probléma olyannyira megoldatlan, hogy az iskolaszámítógépesítési program első hullámában a közép- és általános iskoláknak juttatott több száz, túlnyomórészt magyar gyártmányú iskolaszámítógép a legszükebb – a tv képújság által is használt (!) – magyar karakterhalmazt sem ismeri. Ennek nyelvrontó szerepét ma még nem lehet felmérni, de pedagógusok megfigyelték, hogy ott, ahol van ilyen számítógép, a gyerekek egymással „számítógépül", EKEZET NELKULI NAGYBETUKKEL leveleznek. Lehet, hogy ez csak múló divat, de fel kell figyelni rá.



A probléma megoldatlanságának óriási hátránya, hogy bár ezek a gépek egyéb tulajdonságaik révén alkalmasak, ideálisak lennének a közigazgatásban, a könyvtárakban, a tudományos életben mindennapos nyilvántartási, szövegszerkesztő, szövegfeldolgozó feladatok elvégzésére, használatba vételük csak igen korlátozott lehet, mert a nagy nyilvánosság számára nem készülhetnek olyan nyomtatott szövegek, amelyek nem tartalmazzák a teljes magyar ékezetes betűhalmaz (Á, É, Í, Ó, Ú, Ü, Ű) kis- és nagybetűs változatát.

A pillanatnyi helyzet tehát nagymértékben gátolja a közigazgatás korszerűsítését: az írásmunka támogatásának, a szövegszerkesztésnek, szövegfeldolgozásnak, az irodaautomatizálásnak állít akadályt.

Ezért a Központi Statisztikai Hivatal Számítástechnika-alkalmazási főosztálya a Neumann János Számítógéptudományi Társaság Államigazgatási alkalmazások és Szövegfeldolgozási és humán alkalmazási szakosztályának közreműködésével tudományközi megbeszélést szervez, amelyben eszközgyártók, szoftverfejlesztők, alkalmazók és nyelvészek vesznek majd részt. E fórumon a számítógép-tervezők, gyártók elmondhatják azokat az okokat, amelyek miatt a helyzet így alakult. Megvitathatják, milyen előfeltételeket kellene megteremteni ahhoz, hogy képernyőn és papíron egyaránt, tetszetős kivitelben, a magyar írás szabályainak megfelelő szövegek jelenhessenek meg.

A szoftverkészítők ismertetik majd a probléma programozási vetületét, a képernyő és az írómű jelkészlet-kialakítását, a billentyűzet programozhatóságát, a különböző szűkített képességű íróművek és billentyűzetek programban megoldott rugalmas kezelését, valamint a magyaros szövegszerkesztés követelményeit.

Az alkalmazók a gyakorlatukban felmerülő problémákról, a munkájuk fejlesztéséhez kötődő igényeikről számolnak be. A megbeszélés során feltáruló tapasztalataik értékesen járulhatnak hozzá a megoldandó feladatok pontosabb értelmezéséhez, végleges megfogalmazásához.

A nyelvészek abban segítenek, hogy a gépesíthető írásmunkák különböző területeit érintően kimutatják az élő nyelvben előforduló ékezetes betűk gyakoriságát. Kis- és nagybetűkre vonatkozóan egyaránt megadnak egy-egy sorrendet (prioritási listát), amely irányt mutat a változtatásokhoz. Ennek alapján dönthető majd el, hogy a különböző billentyűzetnagyságú gépek esetében, ha szűkebb a jelhalmaz kerete, mely betűket, jeleket lehet – mert ritkán fordulnak elő – a közvetlenül billentyűzhető jelek közül elhagyni, vagy a billentyűjét szokatlan helyre tenni. Meghatározzák azokat a nyelvi sajátosságokat, amelyek a magyar nyelvű szövegszerkesztésnél gépesíthetők lesznek.

A nyelvészektől várjuk azt is, hogy az összehasonlító nyelvészet eszközeivel is ismertessék a latin betűt sok ékezettel használó szláv és északi nyelvek területén követett gépírási és szövegfeldolgozási gyakorlatot.

A megbeszélés célja olyan dokumentum összeállítása, amely egyeztetett programot ad a probléma országos megoldásához, előkészítve egyúttal az e tárgyú magyar szabványt is. (A megbeszélésről és az eredményként megszülető dokumentumról tájékoztatjuk olvasóinkat. – A szerk.)

SOMFAI GÉZA

Tudja a magyar ábécét!

Az Iskola – számítógép rovat szerkesztőjének megjegyzése a „Környezetunk vedelmerol" és a Pont, pont, vesszőcske..." című cikkekhez.

Az iskolaszámítógép-program keretében először kiosztott HT–1080Z gépek valóban ékezet nélküli betűket írtak, aminek oka kizárólag a gyors programindítás volt. Ez kitűnik abból is, hogy már a korábban vásárolt ABC–80 gépeknél is gondoltak a problémára; a beszerzett gépek nagy része tartalmazott egyes ékezetes betűket (svéd ábécé), valamint abból, hogy a HT-gép második sorozatát szintén ellátták a legfontosabb ékezetekkel.

Hogy a program felelősei tisztában vannak a kérdés jelentőségével, azt az is mutatja, hogy az általános iskolákban mind ez ideig (1984. április) nem került sor géposztásra, aminek oka éppen az „ékezethiány". A TII általános iskolai gépnek csak olyat fogad el, amely a teljes magyar ábécét tudja. Jelenleg egy ilyen ajánlat van, a Microkey. Annyi csak a korlátja, hogy az í, ő, ó, ú és ű csak kisbetűként jeleníthető meg rajta, az összes többi betű kis- és nagybetűs változatával szemben. Előnye azonban, hogy billentyűzete is a magyar szabványt követi.

Mindenesetre ez azt jelzi, hogy a kérdés megoldható. Célszerű lenne természetesen, ha egységes kódrendszer alakulna ki. Ez főleg a – remélhetőleg szintén fejlesztésre kerülő – hasonlóan „magyarul beszélő" mátrixnyomtatók csatlakoztatása szempontjából lenne lényeges. A jelenlegi 7 bites kódszabvány sajnos erre nem elégséges; bővítésére lenne szükség, különös tekintettel a személyi számítógépeknél kialakult 8 bites ASCII(szerű) kódra.

VARGA ANDRÁS

Finom scroll ZX-Spectrumon

Sok játékprogram nélkülözhetetlen eszköze a scroll. Azok azonban, akik nem értenek a gépi kódú programozáshoz, elég nehezen tudnak scrollhoz jutni.

Az itt közölt program olyan speciális scroll, amely csupán egy grafikus jellel viszi balra, illetve jobbra a képernyőt.

JOBBRA:

```
    LD HL, 3FFFH    33,255,63   1
    LD B,C0H        6,192       2
L1: LD C,20H        14,32       3
L2: INC HL          35          4
    RR HL           203,30      5
    DEC C           13          6
    JRNZ L2         32,-6       7
    SCF             55          8
    CCF             63          9
    DEC B           5           10
    JRNZ L1         32,-13      11
    RET             201         12
```

A BALRA program felépítése rendkívül hasonló a jobbra scrollhoz, ezért csak a megváltoztatandó sorokat közöljük.

BALRA:

```
    LD HL, 5800H    33,0,88     1
L2: DEC HL          43          4
    RL HL           203,22      5
```

NÉMETH ATTILA

Inverz kép

ZX-Spectrumon

```
    LD HL, 4ØØØH    33,Ø,64
    LD D, CØH       22,192
LØ: LD B, 2ØH       6,32
L1: LD A, FFH       62,255
    SUB (HL)        15Ø
    LD (HL), A      119
    INC HL          35
    DJNZ L1         16, -7
    DEC D           21
    JRNZ LØ         32, -12
    RET             2Ø1
```

ABC–80-on
(Csak grafikus ábrázolásnál)

```
    LD HL, 7CØØH    33,Ø,124
    LD B, 2ØH       6,32
LØ: LD C, 2ØH       14,32
L1: LD A, 9FH       62,159
    SUB (HL)        15Ø
    LD (HL), A      119
    INC HL          35
    DEC C           13
    JRNZ L1         32,248
    DEC B           5
    JRNZ LØ         32,243
    LD A, Ø8H       62,8
    LD HL, 7CØØH    33,Ø,124
    LD BC, ØØ28H    1,4Ø,Ø
    LD DE, ØØ3ØH    17,48,Ø
L2: LD (HL), 97H    54,151
    ADD, HL, BC     9
    LD (HL), 97H    54,151
    ADD HL, BC      9
    LD (HL), 97H    54,151
    ADD HL, DE      25
    LD (HL), 97H    54, 151
    DEC A           61
    JRNZ L2         32,242
    RET             2Ø1
```

SZABÓ ZOLTÁN,
VERHÁS PÉTER–BÁTORI ENDRE

Ötlet

Hogyan növelhetjük meg a ZX81 képernyőjének nagyságát? Egyszerűen. Ha a programba beírjuk ezt: POKE 16418,Ø, akkor írhatunk a kép vízszintes 22. és 23. sorába.

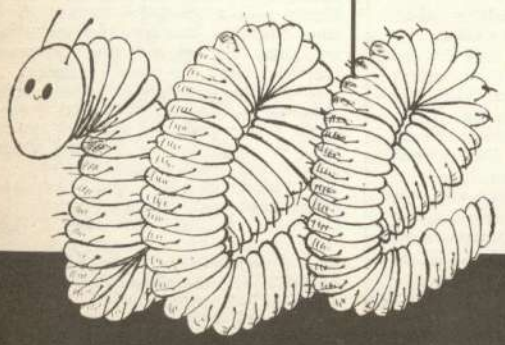
Új függvények bevezetése HT–1080Z-n

Közismert azok számára, akik már láttak EDI leírást, hogy vannak a lemezrendszernek olyan utasításai és függvényei, amelyek újradefiniálhatók. Csakhogy a függvények egy kis problémát okoznak. Hogyan adjuk át a függvényértéket?

Vegyük például az FN függvényt, ami most a 31-es B/K port 15-ös regiszterének tartalmát fogja vizsgálni, és a függvényérték egy 2 bájtos egész szám.

```
FN 4155 JP 7ØØØH
   7ØØØ LD A,ØFH
        OUT (31),A
        IN A,(31)
        PUSH HL
        LD H,Ø
        LDL,A
        LD(4121H),HL      } CALL
            LD A,2        } ØA9AH
            LD (4ØAF H),A }
        POP HL
        JP 1D78 H
```

TASSONYI KADOCSA



lábú

Megállíthatatlan programok ABC–80-on

Néha nem árt, ha illetéktelen személyek nem tudják megállítani programjainkat. Talán sokan nem tudják, hogy ezt el lehet érni. Persze nem egyszerű dolog.

Az ABC–80 gépen minden billentyű lenyomása megszakítást (interruptot) okoz, így a CTX C is. Ezért bármely billentyű (főleg a CTX C) lenyomása lassítja a programfutást. A megszakítást azonban le lehet tiltani. Ehhez gépi kódú programozás szükséges, amihez nem értenek túl sokan.

Mit tegyen egy laikus, ha a kezébe nyomnak egy gépi kódú szubrutint? Ezek egybájtos számsorból állnak. Például:

(1) 243,219,56,38,Ø,111,2Ø1

Ezeket a számokat kell a POKE utasítással elhelyezni a tárban a megfelelő helyre. Ez történhet a 63ØØØ-as címtől így:

(2) POKE 63ØØØ,243,219,56,38,Ø,111,2Ø1

Ezen a tárcímen kezdődik a perifériák kezelésére szolgáló RAM, de perifériákkal nemigen rendelkezünk, így ezek a bájtok parlagon hevernek.

Ezzel a gépi kódú programmal le tudjuk tiltani a CTX C-t, mégis van egy bökkenő! Minden billentyűről való adatbevitel feloldja a letiltást. Nem használhatunk INPUT, INPUT LINE, GET, INP(56) utasításokat. Le kell tehát mondanunk az interaktivitásról? Nem sokat érnénk egy olyan programmal, amely megy magától, nem kérdez semmit, és valahányszor elindítjuk, mindig ugyanazt csinálja.

No, azért nem ilyen sötét a helyzet! Az INP(56) függvényt helyettesíthetjük a szubrutinunkkal. A (2) programsor lefutása után a CALL(63ØØØ) utasítás mint függvény ugyanúgy viselkedik, mint az INP(56), csupán a CTX C gombra nem hajlandó leállni.

Hogyan működik a programunk?

```
(3)  63ØØØ DI              interrupt letiltás
     63ØØ1 IN A, (56)      beolvasás
                           billlentyűzetről
     63ØØ2
     63ØØ3 LD H, ØØ  }     HL regiszterpárba
     63ØØ4           }     töltjük A regiszter
     63ØØ5 LD L, A   }     értékét
     63ØØ6 RET             visszatérés a szub-
                           rutinból
```

A CALL szubrutinhívó utasítás függvényként viselkedik, és értéke az a szám, amely a HL regiszterpárban van a visszatéréskor. A CALL utasításban megadott második argumentum a szubrutin indulása előtt a DE regiszterpárba töltődik.

Ezzel azonban még nem oldódott meg egészen a problémánk, hiszen még mindig nélkülöznünk kell a többi beviteli utasítást. De ezek a szubrutin segítségével mind megírhatók BASIC-ben, csupán türelem és kitartás kell hozzá.

Az ilyen módon védett programok tehát nem állíthatók le csak a gép hátán levő RESET gombbal – akkor viszont a program is elszáll.

VERHÁS PÉTER

Villogásátváltás ABC–80-on

```
      LD HL, 7CØØH    33,Ø,124
      LD,B, 2ØH       6,32
LØ:   LD C, 2ØH       14,32
L1:   LD A, (HL)      127
      ADD A, 8ØH      62,128
      LD (HL), A      119
      INC HL          35
      DEC C           13
      JRNZ L1         32,248
      DEC B           5
      JRNZ LØ         32,243
      RET             2Ø1
```

BÁTORI ENDRE

Fejléc nélküli LOAD-SAVE ZX-Spectrumhoz

A ZX–Spectrum gépen a kimentett adatok elé fejléc kerül, amely tartalmazza az adatok nevét, hosszát és a kezdőcímet. Ha a fejlécet elhagyjuk – BASIC-ben ez nem lehetséges –, akkor a betöltéshez saját LOAD programot kell írni, amely a megfelelő helyre tölti be az adatokat. A betöltést csak az végezheti el, aki ismeri a hossz és a kezdőcím értékét.

```
LOAD            SAVE
SCF             LD (IX),B
LD (IX),B       LD (DE),C
LD (DE),C       LD A, 255
LD A, 255       CALL 1218
CALL 1366       RET
RET
```

„B" jelöli a kezdőcímet, „C" a hosszt.

Ha esetleg két gép áll rendelkezésünkre, e programok segítségével a magnócsatlakozáson keresztül kommunikálhatnak egymással. Természetesen csak rövid memóriarészeket közölhetnek, ha a gyorsaság számít. Ezek a programok a LOAD–SAVE ROM rutinját használják fel, melynek gyorsasága a szalagsebességhez van méretezve. Ha az adatcsere gyorsasága nem számít, akkor a gép felhasználhatja a másik memóriáját például adattárolásra, vagy megfelelő program esetén részfeladatok elvégzésére.

MUDRÁK ISTVÁN

Irodagéptechnikát

Helyszíni javítás az egyik számítóközpontban

Alig három éve, hogy az Irodagéptechnikai Vállalat – az ITV márkanevet változatlanul hagyva – nevet változtatott, és Információtechnikai Vállalat néven működik.

A névváltoztatás tartalmi változásokat is hozott. Az alapító okirat kiegészült valamennyi, az információtechnika fogalomkörébe tartozó tevékenységgel. A hagyományos profilok mellett új, az információtechnika robbanásszerű terjedésével együtt járó technikai eszközök, azok alkalmazástechnikája is a vállalat profiljába kerültek. E rövid áttekintéssel szeretnénk felvázolni az elmúlt rövid időszak fejlődését és azt a perspektivát, amit a vállalat kínál a hazai felhasználók számára.

2400-an a felhasználók szolgálatában

Az ITV „hagyományos" profilját az írógépek, könyvelőgépek, pénztárgépek, másológépek és egyéb alapvetően mechanikus elveken működő gépcsaládok alkották. Ugyancsak „hagyományos módon" ezekre szakosodva alakult ki az országos szervizhálózat, amelynek keretében mintegy 2400 fő állt a felhasználók szolgálatában.

A méretek érzékeltetésére megemlítjük, hogy az ország iroda- és számítógépparkja mintegy 15 milliárd forintot tesz ki, és ennek 73 százalékát az ITV látja el: például éves szinten százezres nagyságrendben javít írógépeket.

A harmincöt év alatt kialakult szakembergárda jó szakmai színvonalat képvisel, amit az is bizonyít, hogy a nagy értékű gépállomány gazdái átalánydíjas, korlátlan időre kötött szerződésekkel biztosítják gépeikre a karbantartást.

Lépést tartva a fejlődéssel

Ahogy az élet gyorsan változik, úgy igyekezett az ITV is lépést tartani, és a számítástechnika térhódításával ez is a vállalati szerviztevékenység része lett.

Felnőtt egy olyan, a számítástechnikához jól értő szakembergárda, amely például az Olivetti vagy a robotron cégnek is méltó partnere lett. Ismerve az alkatrészellátás, az irodai kellékellátás nehézségeit, az ITV újabb lépést tett, és fokozatosan mintegy 15 év alatt felfejlesztette kellék- és alkatrészgyártását. Az előbbit nagyon gyümölcsöző módon az osztrák Kores céggel együttműködve, az utóbbit pedig szélesebb körben (Olivetti, robotron, Videoton stb.). Megállni a gyártásban sem lehetett. Az alkatrészgyártás mellett a nagykőrösi gyártóbázison a Posta Fejlesztési Intézettel együttműködve megindult az utalványbélyegző automaták gyártása, amelyek ma már több százas szériában készülnek az országnak, és exportra is jut belőlük.

Mikroelektronika, mikroszámítógépek

Természetesen a robbanásszerűen fejlődő mikroelektronika is megkívánja a szervizhálózatot, jóllehet a mikroprocesszoros eszközök nagyságrendekkel megbízhatóbbak, mint mechanikus társaik. De az új technikának is vannak gyenge pontjai, és ezek is rendszeres karbantartást igényelnek.

Így kapott helyet a mikroszámítógépes-mikroprocesszoros technika az ITV profiljában, és ma már egyre szaporodik az ebbe a kategóriába tartozó gépek száma. Nem vitás, hogy ez nemcsak a jelen, hanem a jövő is. Fokozatosan minden ITV szerviz profiljában megtalálható szinte valamennyi, a népgazdaságban alkalmazott géptípus. Ezek zöme robotron gép, de örvendetes módon egyre szaporodnak a hazai gyártású, magas színvonalú mikrogépek is. Vállalatunk egyik legfrissebb szerződését a BNV-nagydíjas PROPER gépcsalád szervizellátására kötötte a gyártó-forgalmazó SCL-i-vel.

Az írógéptől a nagyszámítógépig

Az NDK-beli robotron Kombináttal is jó az ITV kapcsolata. A két cég közötti több évtizedes együttműködés eredménye, hogy az egyszerű írógéptől az R 55 M nagyszámítógépig a teljes robotron termékskála szervizét az ITV látja el. A létrehozott konszignációs raktárhálózat hivatott az egyenletes alkatrészellátást biztosítani. (Hogy ez néha döcög? ... Mi sem vagyunk angyalok.)

Szakembereink közül évente mintegy 700-an vesznek részt továbbképző tanfolyamokon. Nem lehet megállni, kétévenként mindenkinek fel kell frissítenie ismereteit! Ezt követeli a felgyorsult technikai fejlődés. Az utóbbi években újabb fogalommal ismerkedtek szervizeink: az alkalmazástechnika kiszélesí-

mikroelektronikáig

Műhelyjavítás, műszeres hibakeresés

tésével, az információtechnika szellemi szolgáltatásainak felfejlesztésével. Ez már napjaink feladata és tulajdonképpen a „hazai mikroszámítógépes társadalom" megteremtéséhez való hozzájárulás – ITV léptékekkel. Hogy ez mit jelent?

Ajánlatot teszünk a megfelelő géptípusra, segítünk beszerzésében, igényeinek megfelelően elkészítjük az alkalmazástechnikát (szervezés – programozás – betanítás), és természetesen a megbízható szervizt is biztosítjuk – ha kell, csereegységekkel, esetleg cseregéppel tesszük lehetővé a folyamatos munkavégzést, ami például főkönyvi zárlati időszakban nem elhanyagolható előny. Aktívan együttműködünk a fejlesztő intézetekkel is.

Lépcsőzetes információfeldolgozási projekt

Felkészülünk a „lépcsőzetes információfeldolgozási projekt" széles körű bevezetésére is. Mit jelent ez?

A hagyományos középgépes könyvelés előnyeit átmentjük korszerű elektronikus könyvelőautomatákra. A kazettán vagy hajlékonylemezen rögzített „másodlagos információt" (az elsődleges a kartonokon, leporellókon közvetlenül megjelenik) ugyancsak a helyszínen üzemeltetett mikroszámítógépen „tovább feldolgozzuk", és olyan információkat nyerünk, amelyek operatívan tájékoztatják a vezetést, a döntést hozókat, de a vállalat minden szintjén dolgozó munkatársakat is. Itt a *gyors, operatív* információn van a hangsúly!

Az összegzett, a teljes vállalatot átfogó havi összesítők elkészítése már nem biztos, hogy a mikroszámítógép feladata. De ami ehhez kell, az tovább feldolgozható adathordozón (kazetta, hajlékonylemez) esetleg adatátviteli vonalon „átlőhető" a nagyobb számítógépre.

Utópia? Nem! Csak éppen ez az irodai automatizálás útja, a gyors, operatív információszolgáltatás egyetlen lehetősége! Ez már „szellemi" gyártás.

Kellékgyártás és -forgalmazás

Visszakanyarodva a hagyományos gyártáshoz: egyre jelentősebb teret kap a mikrogépekhez kapcsolódó kellékgyártás (nyomtatókendők, kazetták, patronok stb.), s a perifériák gyártása is elindul 1984-ben: kazettás data set – elsősorban Commodore mikrogépekhez –, majd irodai mátrixnyomtató bármilyen mikrogéphez. Az igények nagyok, nemcsak itthon, a környező országokban is. S hogy az információtechnika minden területét érintsük, részt veszünk a Posta „szelektív személyi hívórendszer" programjában is. Ennek előnyeit még csak a kísérleti tulajdonosok élvezik. A fejlesztés Magyarországon szokatlan méretű lesz: 1985-ben ezres, 1986–87-ben százezres nagyságrendben kerül az ügyes készülék a felhasználók kezébe.

Európai színvonal

Ha nem is helyettesíti mindenben a telefont, de szolgáltatásaival hazai információtechnikánk egy része európai színvonalúvá válik. S ez nem kis eredmény!

A kereskedelmi hálózatokat is segítjük. Szerelőink a legkorszerűbb elektronikus pénztárgépeket is biztonsággal javítják. És ha már a vonalkód itt van, ennek fogadására is felkészülünk. Az elsőként alkalmazó Skála METRÓ is az ITV-vel kötött szerződést pénztárgépeinek javítására.

Mikroszámítógép-építőknek is

Néhány szót az ITV kereskedelmi szervezetéről, amely szintén nagy fejlődésen ment keresztül. Ma már a kisszámítógépek és a mikrotechnika is megtalálható a kellékek és alkatrészek, elektronikai elemek mellett. A budapesti Flórián áruház mögötti Kerék utcai bolt széles választékával jelentős segítséget nyújt például a mikroszámítógép-építőknek is.

Az ITV szervizhálózat 13 üzem és egy leányvállalat keretében végzi munkáját. A négy budapesti szaküzem egy-egy jelentős szakterületre specializálódva dolgozik, de feladatuk a területi üzemek munkájának segítése is, ahol szintén minden géptípus előfordulhat.

Az információ ma már egyre több gép igénybevételével jut el a feladótól a címzettig, és közben többször átalakul. Az ITV szakemberei tudják, hogy munkájuk nélkül ez a sokmilliárdos eszközállomány holt befektetés lenne. Az ország legnagyobb szervizvállalata ezt nem téveszti szem elől.

Ismeretlen kódok

Ismételten felhívjuk az olvasók figyelmét arra, hogy várjuk egyrészt más mikroprocesszorok „ismeretlen kódjai"-val kapcsolatos tapasztalataikat, másrészt a közöltekre vonatkozó, eltérő tapasztalataik ismertetését.

Az alábbiakban a gyártó által nem közölt 6800 kódokat ismertetjük.

A kódtáblázat „lyukait" itt is táblázatos formában közöljük, de néhány nagyon hatásos vagy érdekes hatású kódra külön is felhívjuk a figyelmet.

A 87, 8F, C7, CF hatására a program módosul, és a programszámláló „ugrik". Ez lehetőséget ad „kiszámított ugrások", tehát akár 65536-felé történő elágazásnak a programba iktatására. (Ez természetesen csak akkor igaz, ha a programnak ez a része a programozható tárterületen van, mert ellenkező esetben a program nem képes önmagát módosítani!) Legyen például a következő programrészletünk:

```
1000 ST2 X
1001 JMP
1002 ab
1003 cd
1004 BRA FB
```

Ennek hatása a következő lesz:

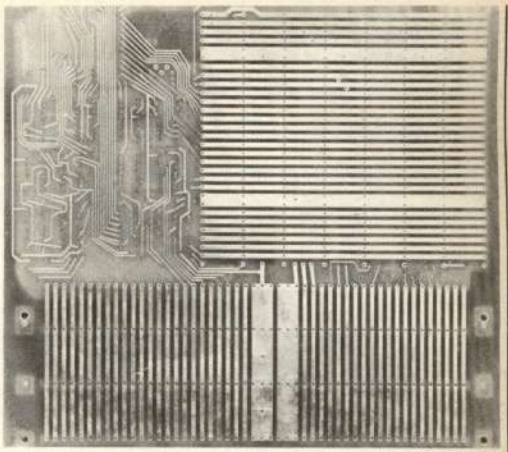
Az indexregiszter tartalma az 1002 és 1003 helyére kerül (az ab és cd a két kitörlendő bájt jelzésére szolgált), és a program az 1004 lépéssel folytatódik. Itt azt találja, hogy ugorj az 1001 lépésre. Ott pedig az áll, hogy ugorj az 1002 és 1003 címek által megadott helyre. Amennyiben az indexregiszterbe előzőleg betöltöttük az elágaztatási számítás eredményét, úgy valóban egy kiszámított ugrást sikerül megvalósítanunk. Ha az 1001 helyen JSR áll, akkor szubrutinokra tudunk hasonlóan ugrani. Az 1000 címen ST2 A utasítást használva, az 1001 helyre feltételes ugró utasítást írva, 1003 helyen kezdve a visszaugrást, és 1004 címre FC értéket töltve, „kiszámított feltételes ugrásokat" tudunk megvalósítani. (Természetesen „csak" 256-felé lehet a programot elágaztatni!)

A 3D, 9D, DD kódok hatására a címbusz folyamatosan nő, a VMA és R/$\overline{W}$ magas értékű, de a mikroprocesszor mégse olvas. Ebből az állapotból csak RESET segítségével lehet kikerülni.

S. E.

UTASÍTÁSNÉV	KÓD	UTASÍTÁSHOSSZ	MEGJEGYZÉS
NOP	00	1	
NOP	02	1	
NOP	03	1	
NOP	04	1	
NOP	05	1	
SBA–1	12	1	A←A−(B+1)
SBA–1	13	1	mint 12
NBA	14	1	A←A AND B
NOP	15	1	
DAA	18	1	nem változtatja a CARRY BIT értékét
ABA	1A	1	mint 18
ABA+1	1C	1	A←A+(B+1), mint 18 és a HALF CARRY BIT sem változik
ABA	1D	1	mint 1C utolsó megjegyzés
TBA	1E	1	
TBA	1F	1	mint 1E, de a CARRY BIT 1
BRA	21	2	
RTS+DES	38	1	mint 39 és 34
RTI+DES	3A	1	mint 3B és 34
DMA	3D	1	lásd szöveg
CPZ A	41	1	mint 4D, de a V és C bitekre is hat
COM A	42	1	
CPY A	45	1	A legalsó bitjét a C bitbe másolja, V = $N\overline{C}+\overline{N}C$, Z = A OR B
DEC A	4A	1	
NOP	4E	1	
CPZ B	51	1	hasonlít 41-hez
COM B	52	1	
CPY B	55	1	hasonlít 45-höz
DEC B	5B	1	
NOP	5E	1	
NEG	61	2	
COM	62	2	
LSR	65	2	
DEC	6B	2	
NEG	71	3	
COM	72	3	
LSR	75	3	
DEC	7B	3	
SBI A#	83	2	A←(A)−(immediate+1)
ST2 A	87	1	lásd szöveg
ST2 S	8F	1	lásd szöveg, csak az SP-vel
SBI A	93	2	A←(A)−(memory direct+1)
DMA	9D	1	lásd szöveg
SBI A X	A3	2	mint 93, csak indexelt
SBI A	B3	3	mint 93, csak abszolút
SBI B#	C3	2	hasonlít 83-hoz
ST2 B	C7	1	hasonlít 87-hez
CPX	CC	3	
BSR	CD	2	
ST2 X	CF	1	lásd szöveg
SBI B	D3	2	hasonlít 93-hoz
CPX	DC	2	
DMA	DD	1	lásd szöveg
SBI B X	E3	2	hasonlít A3-hoz
CPX	EC	2	
JSR	ED	2	
SBI B	F3	3	hasonlít B3-hoz
CPX	FC	3	
JSR	FD	3	

1. kép



2. kép

Építsünk számítógépet! III.

Elöljáróban ismételten kérjük olvasóinkat, hogy további információkért a szerzőhöz forduljanak.

Az első részben ismertetett billentyűzetillesztő egy Optima-típusú írógépbe szerelve látható az *1. képen*. A billentyűzet felett vannak a mechanikai-elektromos átalakítók, a jobb oldali dobozban az ellenállások, míg mellette a kábelcsatlakozó.

A mikroszámítógépek egyik lehetséges csoportosítása annak alapján történik, hogy a gép egyetlen nagy, vagy több, funkciók (funkciócsoportok) szerint elkülönített és egymással információcserét folytató kisebb kártyán valósul-e meg.

Eredetileg mi is egykártyás gépet ígértünk. Szerettük volna az ismertetett gépet csomagban, igen olcsón árusítani. De nem találtunk gyártót! Így terveink módosultak, és a továbbiakban a szerző által használt gépet ismertetjük változatlan formában. Ez a gép nem egykártyás rendszerű, de akként is használható. (Ebben az esetben – a tápegységen kívül még – egy ún. soros vonalon keresztül össze kell kapcsolni egy olyan terminállal, amely a tévévevőhöz és kazettás egységhez csatolja.)

A többkártyás gépek egyes részei közötti információcserét lebonyolító, egy csoportban vezetett vezetékek összességét busznak nevezik. Vannak olyan megoldások, melyeknél az egyes részegységek között csak a legszükségesebb számú vezeték található, és vannak olyanok, amelyeknél minden egyes egység egy közös buszon keresztül kapcsolódik a másikhoz.

Az első megoldás előnye a kis vezetékszám, hátránya, hogy az egyes kártyák csak meghatározott helyen lehetnek, az azonos vezetékszámú kártyacsatlakozókat (a kártyákat egymáshoz, illetve külső egységekhez csatlakoztató rész) a téves csatlakozások elkerülésére eltérővé kell tenni, és a bővítés nehéz. A második megoldáshoz több vezeték kell, de a kártyák bármelyik szabad csatlakozóra csatlakoztathatók a működés zavarása nélkül, a kártyák mérete szabványosítható, a bővítés nagyon egyszerű.

Gépünkben kétféle buszt használunk. Az egyik az univerzális, a másik a perifériák busza *(2. kép)*. A képen látható, hogy egy nagyobb kártyán (az ún. alapkártyán) egyrészt négy, egyenként 50 furatból álló, másrészt erre merőlegesen nyolc, egyenként 30 furatból álló lyuksorozat található.

A kétféle busz jellemzésére szolgál az *1.* és *2. táblázat*. Ezeken az egyes vezetékek (másképpen vonalak) sorszámát, jelölését és rövid leírását adtuk. A kártyák a *3. képen* látható módon csatlakoznak az alapkártyához (és azon keresztül egymáshoz). Ennél a megoldásnál az univerzális buszon a kártyák tetszés szerinti sorrendben helyezhetők el, míg a perifériák buszán nem. Ez utóbbi oka az, hogy ezen a buszon az egyes kártyáknak egy-egy beállítható, de azután már rögzített memória-címcsoport felel meg. Egy perifériaillesztő elhelyezése ezért a működtető programon belül a kártyát kiválasztó cím(ek) módosítását igényli. Mindkét busz (kevés kivételtől eltekintve, melyekre kitérünk) TTL-szintű, digitális jeleket használ.

A buszok vezetékei funkciók szerint csoportosíthatók:

1. Címvezetékek

Az univerzális buszon 20 (9–24.), a perifériabuszon 9 (10–11., 24–27., 30.) vezeték az egyes

1. $\overline{D0}$	Invertált adatvonal
.	.
8. $\overline{D7}$	Invertált adatvonal
9. A15	Címvonal
.	.
24. A0	Címvonal
25. GND	Föld
.	.
27. GND	Föld
28. 7–8 V UNREG	Szabályozatlan +7–8 V feszültségszintű tápvezeték
30. 7–8 V UNREG	Szabályozatlan +7–8 V feszültségszintű tápvezeték
31. −12 V	−12 V szabályozott
32. +12 V	+12 V szabályozott
33. INDEX	„Vakdugó"
34. $\overline{MRST}$	Invertált kézi újraindító
35. $\overline{NMI}$	Invertált feltétel nélküli megszakító
36. $\overline{IRQ}$	Invertált megszakításkérés
37. UD2	Szabadon programozható
38. UD1	Szabadon programozható
39. $\overline{\emptyset 2}$	Invertált adatjel-szinkronizáló órajel
40. $\overline{VMA}$	Invertált érvényes memóriacím-visszajelző
41. R/$\overline{W}$	Olvasás-írás engedélyező
42. $\overline{RESET}$	Invertált gépi újraindító
43. BA	Mikroprocesszor buszról lekapcsolódását jelző
44. Ø1	A másik órajel kiegészítője
45. $\overline{HALT}$	A 43. párja a kérelmező jel
46. A16	Címvonal
.	.
49. A19	Címvonal
50. BAUD RATE	Periféria (soros) órajel

1. táblázat. Univerzális busz

egységek, részegységek, memóriahelyek kiválasztására (megcímzésére) szolgál. A 20 vezeték összesen 1 Mbájt nagyságú memória megcímzését teszi lehetővé azzal, hogy a 20 vezeték melyikén van logikai 1, illetve 0 állapot. (A kiválasztás a megfelelő hely(ek)en levő logikai 1 értékkel történik.) Amennyiben a perifériabusz összes vezetékét egy-egy perifériaillesztő kiválasztására használjuk, úgy 512 perifériaillesztőt választhatunk ki. (Ez, ahogy látni fogjuk, sokkal több perifériát is jelenthet, mert például egy illesztő 16 analóg bemenetet vagy 4 lemezhajtót jelenthet. Ennyi illesztő, ahogy a képeken is látható, nem helyezhető el az alapkártyán. Az alapkártya méreteinek „kiterjesztésére" szolgál a buszkiterjesztő. Ez egy olyan kábel, amely a buszvezeték számával azonos számú vezetékből áll, és amelynek mindkét végén kártyacsatlakozó van. Ezzel több alapkártyát lehet összekötni.)

2. Adatvezetékek

Mindkét buszon 8 adatvezeték (a főbuszon 1–8., a perifériabuszon 12–19.) található (a főbuszon invertált – a logikai 1 az alacsonyabb TTL-szint). A kiválasztott címek között ezeken keresztül történik az információáramlás.

3. Tápvezetékek

A számítógép közösített föld- és különböző feszültségszintű tápvezetékei (a főbuszon 25–32., a perifériabuszon 3–6., 22–23.). Ezek nem TTL-szintűek. A föld a viszonyítási alap, a 0 V; a fő tápvezetékben +7–8 V szabályozatlan feszültségű áram folyik, amelyből minden egyes kártyán külön-külön történik a működtető +5 V feszültség előállítása. (Ennek a megoldásnak előnye, hogy a szabályozásnál fejlődő hő szétoszlik, és csak annyi energia kerül szabályozásra, amennyi beépített kártya van. Hátránya, hogy a kártyákon a feszültségszabályozó aránylag jelentékeny helyet foglal el.) Az ún. RS–232C szabvány szerinti vonalak kiszolgálásához a buszokon +12 és −12 V is rendelkezésre áll.

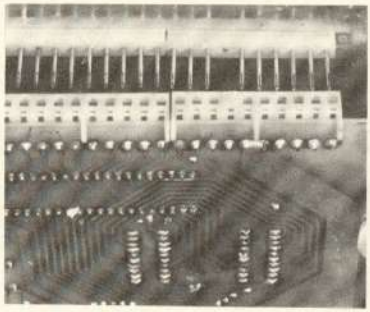
4. INDEX

Ez egy csatlakoztatást kizáró vonal (*4. kép*, a főbuszon a 33., a perifériabuszon a 7.). A képen látható, hogy ezzel a módszerrel kizárható a kártyák hibás helyzetű (fordítva vagy eltolva) csatlakoztatása. Ez nagyon fontos, egyszerű és hatásos buszvédelem.

1. UD3	mint főbusz 37.
2. UD4	mint főbusz 37.
3. −12 V	mint főbusz 31.
4. +12 V	mint főbusz 32.
5. GND	mint főbusz 25.
6. GND	mint főbusz 25.
7. INDEX	mint főbusz 33.
8. $\overline{\text{NMI}}$	mint főbusz 35.
9. $\overline{\text{IRQ}}$	mint főbusz 36.
10. RS0	Perifériakiválasztó címvezeték
11. RS1	Perifériakiválasztó címvezeték
12. D0	Adatvonal
.	.
19. D7	Adatvonal
20. Ø2	mint főbusz 39., de nem invertált
21. R/$\overline{\text{W}}$	mint főbusz 41.
22. +7–8 V	mint főbusz 28.
23. UNREG	mint főbusz 28.
24. A16	mint főbusz 46.
.	.
27. A19	mint főbusz 49.
28. BAUD RATE	mint főbusz 50
29. $\overline{\text{RESET}}$	mint főbusz 42.
30. –	Be nem kötött perifériakiválasztó címvezetékek (3)

2. táblázat. Perifériabusz

3. kép



5. Memóriavezérlő vezetékek

Ide tartoznak az órajelek (a főbuszon a 39. invertált és a 44., a perifériabuszon a 20.), amelyek a szinkron működést vezérlik, a $\overline{\text{VMA}}$ (a főbusz 40. vezetéke, ami az érvényes memória működését jelzi), az R/$\overline{\text{W}}$ (a főbuszon a 41., a perifériabuszon a 21., ami az adatforgalom irányát – olvasás vagy írás – jelöli ki.

6. Külső vezérlést irányító vezetékek

A mikroprocesszor működését megszakítók (a főbuszon a 34. és a 42., a perifériabuszon a 29. az újraindítást kezdeményezők; a főbuszon a 35., a perifériabuszon a 8. a feltétel nélküli megszakítást kiváltók; a főbuszon a 36. és a perifériabuszon a 9. a feltételes megszakítást igénylők; a főbuszon a 45. és 43. a mikroprocesszort leállítók, illetve a buszról leválasztók).

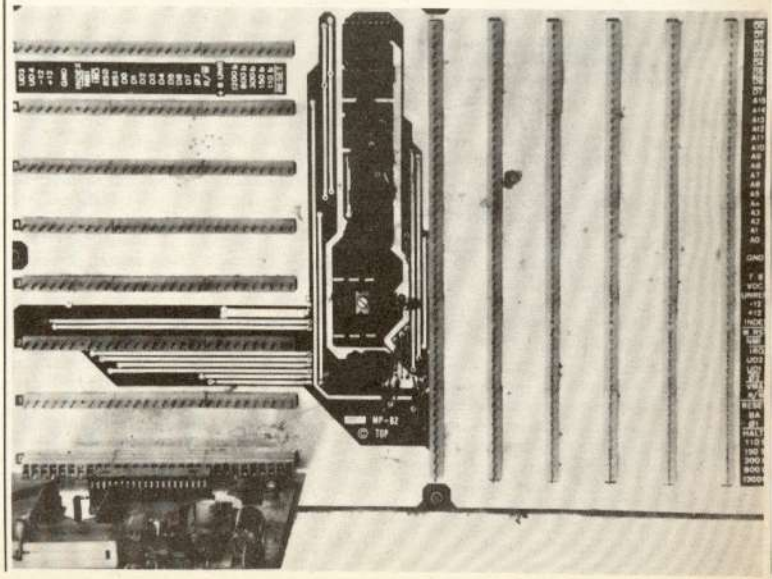
7. Perifériaműködést kiszolgálók

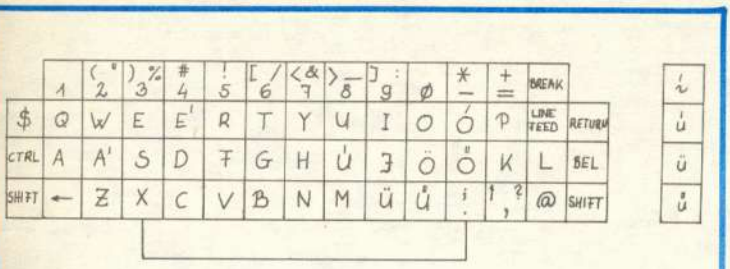
A már említett +12 és −12 V feszültségeken felül a Baud-sebességnek nevezett órajel (a főbuszon az 50., a perifériabuszon a 28.) és a két szabadon meghatározható vonal (a főbuszon a 37–38., a perifériabuszon az 1–2.).

Ezek között vannak, amelyeket minden kártya használ (például a cím- és adatvezetékek), amelyeket csak egy-egy kártya (például a Baud-sebesség), és van, amelyet az esetleges speciális használatra foglaltak le (például a BA, $\overline{\text{HALT}}$). Az utóbbiakat (a központi egység kártya módosításával) esetleg más célra is használhatjuk (például növelni lehet az adat- vagy címvonalak számát). Ebben az esetben azonban a módosítatlan buszhoz készült kártyák esetleg nem, vagy hibásan fognak működni a módosított buszon.

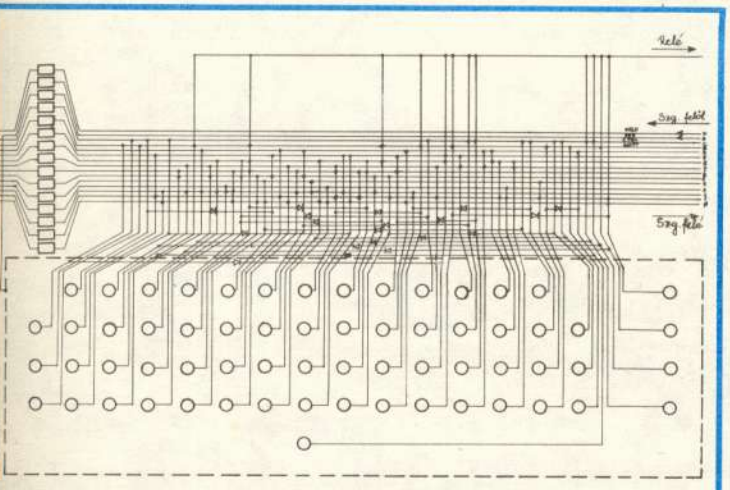
DR. SIMONYI ENDRE

4. kép





A billentyűzet



A billentyűzet áramköreinek elve, kapcsolási rajza

Mi már építünk!

A Szerkesztőség kérésének eleget téve, közlöm a sorozat gépének építésével kapcsolatos tapasztalatainat.

Először magamról: a Landler Jenő Gimnázium harmadik osztályos tanulója vagyok. Még tavaly megismerkedtem az iskolaszámítógéppel, de csak az idén kezdtem komolyabban foglalkozni számítógépekkel. Az az ötlet, hogy számítógépet építsek – bár semmiféle ilyen irányú képzettségem nincs –, a µ M-nak köszönhető. Itt olvastam a HCC-ről és arról, hogy a gépépítők között diákok is vannak.

Első lépésként a sorozat szerzőjénél érdeklődtem, majd megpróbáltam a suliban társakat találni. Nem sikerült, mert a legtöbb kapott kész gépet a szüleitől. Az ELTE fizika-előkészítőjén találtam egy társat, de aztán ő is abbahagyta.

Mindenekelőtt a billentyűzetet kellett elkészítenem, mert csak ezután érdemes hozzákezdeni a gép építéséhez. Három lehetőség között választhattam: 1. veszek egy kész billentyűzetet; 2. átalakítok egy írógépet; 3. magam barkácsolok egyet.

Én az utolsót választottam. Az írógép-változatból kiindulva készítettem el a kapcsolási rajzot és a billentyűzet sémáját (ezek a mellékelt ábrán láthatók). Az igazi problémát a billentyűzet mechanikája és az érintkezők okozták. Szerencsére sikerült olyan selejt érintkezőket beszerezni, amelyek erre a célra még megfeleltek.

Lehetőségem adódott arra, hogy az érintkezőket NYÁK-ra építsem rá. Osztályfőnökömön, Szabó Béla segítségével el is készült a kártya. Az érintkezők beforrasztásán már túl vagyok, most jön az élesztés. Erről majd egy későbbi számban írok.

GATTYÁN ANDRÁS

COMMODORE 64

Egy BASIC editor hiba

A BASIC editor üzemmód a BASIC programsorok bevitelére, javítására szolgál. Ennek az üzemmódnak a kezdetét a mindenki számára ismerős „READY" üzenet jelzi.

Programsorok javítása közben gyakran előfordul, hogy a CURSOR UP billentyűvel fel kell léptetni a cursort a javítandó sorig, és a javítás után a CURSOR DOWN billentyűvel leléptetni a legalsó sorba – esetleg egyúttal egy ROLL funkciót is kiváltva.

Ha ezután leírjuk a következő programsort, amely például egy elírás miatt hosszabb, mint a megengedett 80 jel, nagyon veszélyes helyzet áll elő.

Ha a felesleges jeleket a DEL törlőbillentyűvel töröljük, a 81. jel törlésekor a tárban lévő program „magától" elindul.

A program az első INPUT, GET vagy hibás utasításig lefut, a megfelelő üzenet a képernyőn megjelenik, de a billentyűzet – még a STOP és a RESTORE billentyű is – hatástalan marad. Az operációs rendszer egy hibás gépi ciklusba került, amelyből látszólag csak a gép kikapcsolásával lehet kimenteni. Ez esetben természetesen a tárban levő program – és vele a felvitelre, ellenőrzésre fordított munka – elvész. Ennek a hibának a jelentkezésére a dokumentáció nem utal, a fenti eljárást nem tiltja.

Kizárólag kísérleti úton találtam egy – bár nem egyszerű – módszert, amelynek segítségével az operációs rendszer a hibás ciklusból kiemelhető a tártartalom elvesztése nélkül, ha a program hibaüzenettel félbeszakadt, vagy egy GET (és nem INPUT) utasításon várakozik.

A teendők a következők. Le kell nyomni a SHIFT, a SPACE (szóköz), majd a „3" jelű billentyűt (egyszerre nyomva tartani mindhármat!), majd felengedni a „3" jelűt. Ekkor a képernyőn a PRESS PLAY ON TAPE üzenet jelenik meg. A STOP még mindig hatástalan. Le kell nyomni a magnetofon „lejátszás" gombját – természetesen előtte ki kell emelni a kazettát. A képernyő előírásszerűen elveszti a tartalmát, a magnó motorja forogni kezd.

Ebben a fázisban már eredményesen használható a STOP billentyű. A gép kikerült a hibás ciklusból, és a továbbiakban – egy ajánlatos STOP-RESTORE után – folytatható a program felvitele.

A fenti hiba és bonyolult (nem is mindig lehetséges) feloldása elkerülhető, ha a DEL törlőbillentyű használata helyett a SHIFT és a RETURN billentyű használatával lezárjuk a 80 jelnél hosszabb sor bevitelét, majd a kérdéses sort ismét bevisszük – jobban ügyelve a hosszára.

ZELCZER GYÖRGY

OSAK

ORSZÁGOS SZOFTVER ARCHÍVUM ÉS KÖVETŐSZOLGÁLAT AJÁNLATA

PRIZMA MODUL

A COMMODORE–64 számítógépet alkalmassá teszi a Z80 alapú mikrogépeken működő CP/M operációs rendszerbe írt programok változtatás nélküli futtatására. A szokásos lehetőségeken túl támogatja 2 db floppy-meghajtó vezérlését, valamint kiszolgálja a V–24 interface-t.

16 800,– Ft

PRIZMA BASIC

A COMMODORE–64 számítógép BASIC utasításkészletét mintegy 70 új utasítással bővíti. A leglényegesebb új funkciók: magyar betűkészlet, képernyő vezérlés, programirást segítő funkciók, számváltások, grafikakezelés, hang generátor programozása.

11 500,– Ft

SOS OSAK–01

A COMMODORE–64 számítógépen futó rendszer a programok tesztelését 16 új funkcióval támogatja, többek között: automatikus átsorszámozás, decimális szám átalakítása, sortörlés, szövegszerkesztés, nyomkövetés, lépésenkénti végrehajtás, automatikus ismétlés.

12 600,– Ft

SOS OSAK–02

A COMMODORE–64 számítógépen futtatható programrendszer 17 új utasítással segíti elő a programozási munkákat, többek között: assembler írás, disassembler, nyomtató kikapcsolása, keresés a tárban, képernyő törlése.

12 600,– Ft

ÉRDEKLŐDÉS – MEGRENDELÉS

SZÁMALK

SZÁMALK-OSAK Önálló Főosztály: Bp. XI., Vahot u. 6. Tel.: 669-156
SKÁLA-PRIZMA Áruház: Bp. X., Gyakorló köz 2–6.
SZÁMOK Könyvesbolt: Bp. XI., Szakasits Árpád út 68.

Ígéretünkhöz híven folytatjuk a játékprogram-pályázaton díjat nyert programok közlését. A listák elemzését mindazoknak ajánljuk, akik a jövő évi versenyen indulni szeretnének, melynek kiírását ugyancsak e számunkban, a 4. oldalon olvashatják.

Bruckner Nándor, a második díjas

Rátkai István, harmadik díjas

Aki mindent leáratott: Tóth Péter, játékprogram-pályázatunk nyertese, és a közönség első számú helyezettje

A második díjas játékprogram

SIKER VALÓBAN

Bruckner Nándor ezt a jeligét adta programjának. A rövid lista begépelése megéri a fáradságot, mert nagyon ügyes, új kígyós játék lesz belőle. A program HT-gépre készült

```
10 CLS:PRINT TAB(28)"FUTKOSO":PRINT@2*64+1,"A KIGYOT ELINDULASA UTAN AT KELL JUT
TATNI A KULONBOZO AKADALYO-KON. A KIGYOT A JOBB FELSO SAROKBAN TALALHATO JEL BEL
SO CSUCSKE-BE KELL ELVEZETNI."
20 PRINT TAB(9) "A KIGYOT A MENETIRANY SZERINTI JOBB OLDAL FELE
 IRANYITJUK A ";CHR$(34);"J";CHR$(34);" GOMBBAL."
30 PRINT TAB(9) "A MENETIRANY SZERINTI BAL OLDAL FELE IRANYITJUK
 A ";CHR$(34);"B";CHR$(34);" GOMBBAL."
40 PRINT TAB(5) "A KIGYOT TEHAT UGY IRANYITJUK, MINTHA A FEJEBEN ULNENK."
50 PRINT:PRINT TAB(10)"MELYIK NEHEZSEGI FOKOT VALASZTJA? (1,2,3)"
60 A$=INKEY$:IFA$="1" THEN Z=5  ELSE IF A$="2" THEN Z= 2 ELSE IF A$="3" THEN Z=0
 ELSE 60
70 CLS:FOR I=0 TO 127:SET(I,0):SET(I,47):NEXT
80 FOR J=0 TO 47:SET(0,J):SET(127,J):NEXT
90 FOR I=0 TO 125 STEP RND(4)+4
100 FOR K=I TO I+2:SET(K,9):SET(K,29):NEXT:NEXT
110 FOR I=0 TO 125 STEP RND(4)+4
120 FORK=I TO I+2:SET(K,19):SET(K,39):NEXT:NEXT
130 FOR J=0 TO 47 :SET(40,J):SET(80,J):NEXT
140 SET(121,5):SET(121,5):SET(121,4):SET(120,4):SET(119,4)
150 I=1:J=44:SET(I,J):U=1:V=0:Y=20+RND(27)
160 OUT31,7:OUT30,254:OUT31,1:OUT30,0:OUT31,8:OUT30,15
170 FOR X=0 TO Z:A$=INKEY$:IF A$="B" THEN IF U<>0 THEN V=-U:U=0 ELSE U=V:V=0
180 IF A$="J" THEN IF U<>0 THEN V=U:U=0 ELSE U=-V:V=0
190 IF A$<>"J" AND A$<>"B" THEN NEXT
200 SET(40,Y):RESET(40,Y-5):SET(80,Y):RESET(80,Y-5)
210 Y=Y-1:IF Y<6 THEN Y=47:GOTO 260
220 I=I+U:J=J+V:OUT 31,0:OUT30,20+I+J
230 A=POINT(I,J):IF A=-1 THEN 270 ELSE SET(I,J)
240 IF I=120 AND J=5 THEN 300
250 GOTO 170
260 FOR L=5 TO 0 STEP-1:SET(40,L):SET(80,L):NEXT:GOTO 220
270 FOR B=20+I+J TO 255:OUT31,0:OUT30,B:NEXT:OUT31,7:OUT30,255
280 PRINT@14+64+44,"FOLYTATJA (I v N)?";
290 F$=INKEY$:IF F$="I" THEN 10 ELSE IF F$="N" THEN CLS:PRINT@5+64+20,"REMELEM J
OL SZORAKOZOTT !":PRINT@12+64,:END ELSE 290
300 B=20+I+J
310 OUT 31,0:OUT30,B:OUT31,0:OUT30,B-10:B=B-1:IF B>50 THEN 310
320 OUT31,0:OUT30,B+50:OUT31,7:OUT30,255
330 GOTO 280
```

A harmadik díjas játékprogram

TIE

Rátkai István a Csillagok háborújának TIE-vadászok elleni harcába veti be a játékost. Állandó támadások közepette kell az irányzékot beállítani és megsemmisíteni a vadászokat. A jó reflexeket kívánó program ABC–80-ra készült.

```
10 T6=1
20 FOR I=1 TO 8 : B1(I)=3000 : B$(I)="          " :
   NEXT I
30 RANDOMIZE : P=1
40 M7=3000
50 X%=RND*40+20 : Y%=RND*40+20 : W5%=0
60 RESTORE
70 ? CHR$(12)
80 REM A celpont koordinatainak megadasa.
90 DIM A%(23),B%(23)
100 DATA 1,1,2,2,2,2,2,3,3,3,3,3,3,3,3,3,4,4,4,4,
   4,5,5,
110 FOR I%=1% TO 23% : READ A%(I%) : NEXT I%
120 DATA 2,8,1,4,5,6,9,1,2,3,4,5,6,7,8,9,1,4,5,6,
   9,2,8,
130 FOR I%=1% TO 23% : READ B%(I%) : NEXT I% : W%
   =0%
140 IF T6=1 THEN 1790
150 ? "Kersz tajekoztatot?"; : GET L$
160 GET U$
170 IF U$="I" THEN 190
180 IF U$="N" THEN 330 ELSE ? "NEM ERTEM!!!" : GO
   TO 160
190 ? CHR$(12)
200 ? "Ez a program a 'CSILLAGOK HABORUJA'      al
   apjan keszult.Te egy X-szarnyu"
210 ? "pilotaja vagy.Celodia kepernyo"
220 ? "kozepere hoznod es ott megsemmisiteni     a
   kepernyon feltuno TIE-vadaszokat."
230 ? "urhajodat a Z,A,Ö es a ´ billentyu-"
240 ? "vel iranyithatod.A SPACE-szel lohetsz.  VI
   GYAZZ!!Egy masik TIE-vadasz tesed"
250 ? "kovetes ha lassu vagy,ro lo ki"
260 ? "tesed!!!"
270 ? "Ennek hozzad viszonyitott helyzetet a "
280 ? "kepernyo aljan es Jobb oldalan lathatod,ba
   l oldalon pedig az energiad."
290 ? "Uj Jatekot P-vel (Play) kerhetsz."
300 ? "U.I:VIGYAZZ!Ez a program szimulator      Pr
   ogram!"
310 ? "SOK SZERENCSET!                          NY
   OMJ MEG EGY BILLENTYUT!"
320 GET F$
330 ? CHR$(12)
340 FOR I%=0% TO 23% : ? : ? CHR$(151%); : NEXT I
   % : ? CHR$(135);
350 SETDOT 36,34 : SETDOT 36,46 : SETDOT 30,40 :
   SETDOT 42,40
360 FOR I=3 TO 78 : SETDOT 4,I : SETDOT 68,I : NE
   XT I : CLRDOT 68,3 : CLRDOT 69,3
370 FOR I=4 TO 71 : SETDOT I,2 : SETDOT I,4 : SET
   DOT I,75 : NEXT I
380 SETDOT 37,78 : SETDOT 36,78 : SETDOT 35,78
390 SETDOT 71,39 : SETDOT 71,40 : SETDOT 71,41
400 REM Az 'ON P GOTO'a Jatek es a demon      . s
   tracio kozti kulonbsegek miatt          van.
410 ON P GOTO 420,2010
420 POKE 31744,127 : POKE 31747,127 : POKE 31758,
   127 : POKE 31770,127 : POKE 31780,127 : POKE 3
   1783,127
430 ? CUR(0,4);"PONT:" : ? CUR(0,15);"HATRAVAN:"
   : ? CUR(0,27);"ELMENT:"
440 FOR M%=20% TO 1% STEP -1% : K1%=41%
450 FOR I=4 TO 71 : SETDOT I,2 : NEXT I
460 A%=RND*40+20% : B%=RND*40+20% : A2%=0% : B2%=
   0% : A3%=1% : B3%=1%
470 ON P GOTO 480,490
480 OUT 6,207 : FOR I=1 TO 2000 : NEXT I : OUT 6,
   0
490 K1%=K1%+1% : S2%=S2%+1 : IF K1%>710% THEN 157
   0 ELSE CLRDOT K1%/10%,2% : CLRDOT (K1%-1%)/10%
   ,2%
500 ON P GOTO 510,620
510 A5%=ABS(39%-A%) : B5%=ABS(45%-B%)
520 OUT 6,17
530 REM A kulonbozo ertekek kinyomtatasa
540 IF A5%<10 THEN ? CUR(0,0);" ";A5% ELSE ? CUR(
   0,0);A5%
550 IF B5%<10 THEN ? CUR(0,36);" ";B5% ELSE ? CUR
   (0,36);B5%
560 ? CUR(0,8);W%
570 IF M%<10 THEN ? CUR(0,24);M% : GOTO 590
580 ? CUR(0,23);M%
590 ? CUR(0,33);W5%
600 POKE 31744,127 : POKE 31747,127 : POKE 31758,
   127 : POKE 31770,127 : POKE 31780,127 : POKE 3
   1783,127
610 REM A TIE-vadasz kirajzolasa
620 FOR I%=1% TO 23% : CLRDOT A%+A%(I%),B%+B%(I%)
   : NEXT I%
630 REM ATIE-vadasz mozgatasa
640 IF A2%<A3% THEN A1%=RND*3%-1% : A2%=RND*40%+2
   0% : A3%=1%
650 A3%=A3%+1% : A%=A%+A1%
660 REM A koveto TIE-vadasz helyzetenek          t
   orlese
670 CLRDOT 69%,X% : CLRDOT 70%,X%
680 CLRDOT Y%,76% : CLRDOT Y%,77%
690 ON P GOTO 700,2020
700 IF INP(56)=193% THEN A%=A%+2% : Y%=Y%+2%
710 IF INP(56)=218% THEN A%=A%-2% : Y%=Y%-2%
720 IF A%<4% OR A%>62% THEN W5%=W5%+1% : GOTO 154
   0
730 IF B2%<B3% THEN B1%=RND*3%-1% : B2%=RND*40%+2
   0% : B3%=1%
740 B3%=B3%+1% : B%=B%+B1%
750 ON P GOTO 760,790
760 IF INP(56)=219% THEN B%=B%+2% : X%=X%+2%
770 IF INP(56)=167% THEN B%=B%-2% : X%=X%-2%
780 REM Koveto TIE-vadasz mozgatasa
790 IF X%=40% THEN 810 : H%=0%
800 IF X%<40% THEN X%=X%+1% ELSE X%=X%-1%
810 IF X%<5% THEN X%=6%
820 IF X%>74% THEN X%=74%
830 IF Y%=36 THEN 850 : H1%=0%
840 IF Y%<36% THEN Y%=Y%+1% ELSE Y%=Y%-1
850 IF Y%<5% THEN Y%=6%
860 IF Y%>67% THEN Y%=67%
870 H%=H%+1% : H1%=H1%+1% : IF H%>4% AND H1%>4% T
   HEN 990
880 ON P GOTO 890,900
890 IF B%<5% OR B%>65% THEN W5%=W5%+1% : GOTO 154
   0
900 FOR I%=1% TO 23% : SETDOT A%+A%(I%),B%+B%(I%)
   : NEXT I%
910 REM Celkereszt kirajzolasa
920 SETDOT 36,34 : SETDOT 36,46 : SETDOT 30,40 :
   SETDOT 42,40
930 REM Koveto TIE-vadasz helyzetenek          k
   irajzolasa
940 SETDOT 69%,X% : SETDOT 70%,X%
950 SETDOT Y%,76% : SETDOT Y%,77%
960 ON P GOTO 970,2120
970 IF INP(56)=160% THEN 1370
980 GOTO 490
990 GOTO 1580
1000 ? CHR$(12)
1010 ? "Ugy velem kilottek!" : ? "Hat nem figyelm
   eztettelek!" : ? "Eredmenyed:";W%;" pont"
1020 ? "MEG";M%;" TIE-VADASZ JOTT VOLNA"
1030 IF W%=0 THEN ? "GYER VAGY TE EHHEZ!!!"
1040 IF W%>B1(8) THEN ? " Ird be a neved!(8 betut
   !)"; ELSE 1100
1050 GET L$
1060 INPUT B$
1070 REM 'EREDMENY-TOMB'sorbarendezese
1080 FOR I=7 TO 1 STEP -1 : B1(I+1)=B1(I) : B$(I+
   1)=B$(I) : IF B1(I)<W% THEN NEXT I
1090 B1(I+1)=W% : B$(I+1)=B$
1100 REM EREDMENY-TABLAZAT Kiirasa
1110 ? CHR$(12)
```

```
1120 FOR I=1 TO 8 : ? CUR(7+I,12);I;". ";0$(I);CU
     R(7+I,24);":  ";D1(I);"                    "
1130 NEXT I
1140 FOR I%=0% TO 6% : ? CUR(I%,0%);CHR$(151) : N
     EXT I%
1150 FOR I%=4% TO 16% : SETDOT 3%,I% : SETDOT 3%,
     I%+20% : SETDOT 18%,I%+20% : SETDOT 11%,I%+20%
     : NEXT I%
1160 CLRDOT 11%,33% : CLRDOT 11%,34% : CLRDOT 11%
     ,35% : CLRDOT 11%,36%
1170 FOR I%=3% TO 18% : SETDOT I%,10% : SETDOT I%
     ,20% : SETDOT I%,24% : NEXT I%
1180 ? CUR(5%,22%);"EREDMENY TABLAZAT"
1190 FOR I%=43% TO 78% : SETDOT 13%,I% : SETDOT 1
     8%,I% : NEXT I%
1200 FOR I%=14% TO 17% : SETDOT I%,43% : SETDOT I
     %,78% : NEXT I%
1210 ? CUR(20,14);"*P-vel indithatsz*"
1220 FOR I=1 TO 120
1230 ? CUR(19,14);"*  *  *  *  *  *  *"
1240 ? CUR(21,14);"  *  *  *  *  *  *"
1250 FOR F9=1 TO 50 : NEXT F9
1260 ? CUR(19,14);" *  *  *  *  *  *  *"
1270 ? CUR(21,14);" *  *  *  *  *  *  *"
1280 FOR F9=1 TO 50 : NEXT F9
1290 ? CUR(19,14);"  *  *  *  *  *  *"
1300 ? CUR(21,14);"*  *  *  *  *  *  *"
1310 FOR F9=1 TO 45 : NEXT F9
1320 IF INP(56)=80 THEN 30 ELSE NEXT I : ? CHR$(1
     2) : GOTO 1790
1330 ? CHR$(12)
1335 ? "Elfogytak a TIE-vadaszok!" : ? W5%;" mesl
     osott!                                Eredmenyed
     :";W%;" p"
1340 IF W%>10000% THEN ? "MENJ EL VADASZPILOTANAK
     !!!"
1350 GOTO 1040
1360 REM LOVES
1370 F1%=6%
1380 FOR F=28% TO 35% STEP 1/2
1390 SETDOT F,F1% : SETDOT F,80%-F1% : SETDOT 72%
     -F,F1% : SETDOT 72%-F,80%-F1%
1400 FOR I=1 TO 10 : NEXT I
1410 CLRDOT F,F1% : CLRDOT F,80%-F1% : CLRDOT 72%
     -F,F1% : CLRDOT 72%-F,80%-F1%
1420 F1%=F1%+2%
1430 NEXT F
1440 K1%=K1%+10%
1450 IF DOT(36,40) THEN 1470 ELSE 980
1460 REM ROBBANAS
1470 IF P=1 THEN OUT 6,73
1480 FOR I%=1% TO 23% : CLRDOT A%+A%(I%),B%+B%(I%
     ) : NEXT I%
1490 SETDOT 36,46 : SETDOT 30,40 : SETDOT 36,34 :
     SETDOT 42,40
1500 FOR I%=1% TO 20% : SETDOT 36%+I%,40%+I% : SE
     TDOT 36%+I%,40%-I% : SETDOT 36%-I%,40%+I% : SE
     TDOT 36%-I%,40%-I%
1510 FOR D=1 TO 10 : NEXT D
1520 CLRDOT 36%+I%,40%+I% : CLRDOT 36%+I%,40%-I%
     : CLRDOT 36%-I%,40%+I% : CLRDOT 36%-I%,40%-I%
     : NEXT I%
1530 FOR I=1 TO 700 : NEXT I : OUT 6,0 : W%=W%-K1
     %+720%
1540 NEXT M%
1550 GOTO 1330
1560 END
1570 ? CHR$(12) : ? "Elfogyott az energiad!
                  Eredmenyed:";W%;" pont."
1575 OUT 6,0 : FOR I=1 TO 5000 : NEXT I : GOTO 10
     40
1580 IF P=1 THEN OUT 6,73
1590 T1=5% : T2=40%
1600 REM Jatekos robbanasa
1610 FOR T%=36% TO 66% : T2=T2+T1/10% : SETDOT T%
     ,T2 : NEXT T%
1620 T1=8% : T2=36%
1630 FOR T%=40% TO 74% : T2=T2+T1/10% : SETDOT T2
     ,T% : NEXT T%
1640 T1=2% : T2=36%
1650 FOR T%=40% TO 74% : T2=T2-T1/10% : SETDOT T2
     ,T% : NEXT T%
1660 T1=1% : T2=36%
1670 FOR T%=40% TO 10% STEP -1 : T2=T2+T1/10% : S
     ETDOT T2,T% : NEXT T%
1680 T1=1% : T2=40%
1690 FOR T%=36% TO 8% STEP -1 : T2=T2+T1/10% : SE
     TDOT T%,T2 : NEXT T%
1700 T1=7% : T2=40%
1710 FOR T%=36% TO 6% STEP -1 : T2=T2-T1/10% : SE
     TDOT T%,T2 : NEXT T%
1720 T1=4% : T2=40%
1730 FOR T%=36 TO 67 : T2=T2-T1/10% : SETDOT T%,T
     2 : NEXT T%
1740 FOR Y=1 TO 1000 : NEXT Y
1750 OUT 6,0
1760 IF P=2 THEN 1110
1770 GOTO 1000
1780 REM TIE kirajzolas
1790 FOR I%=0% TO 23% : ? : ? CHR$(151%); : NEXT
     I%
1800 T6=2
1810 FOR I%=0% TO 29%
1820 FOR I1%=0% TO 3%
1830 SETDOT 10%+I1%,2%+I%
1840 SETDOT 10%+I1%,77%-I%
1850 SETDOT 50%+I1%,77%-I%
1860 SETDOT 30%+I1%,77%-I%
1870 SETDOT 10%+I%,15%+I1%
1880 SETDOT 53%-I%,15%+I1%
1890 SETDOT 10%+I%,38%+I1%
1900 SETDOT 53%-I%,38%+I1%
1910 SETDOT 10%+I%,48%+I1%
1920 SETDOT 53%-I%,48%+I1%
1930 NEXT I1%
1940 IF I%>=22% THEN FOR I1%=0% TO 3% : CLRDOT 30
     %+I1%,100%-I% : NEXT I1%
1950 NEXT I%
1960 ? CUR(21,0);"       Made by RATKAI ISTVAN  MC
     MLXXXIII";
1970 ? CUR(23,0);"                P-vel indithatsz"
1980 FOR I=1 TO 2500 : IF INP(56)=80 THEN 30 ELSE
     NEXT I
1990 P=2 : GOTO 340
2000 REM Demonstracios elteresek
2010 ? CUR(0,0);" P-vel indithatsz    RECORD:";D1(
     1) : GOTO 440
2020 IF A%=33% THEN 2050
2030 IF A%<33% THEN A%=A%+2 : Y%=Y%+2
2040 A%=A%-2 : Y%=Y%-2 : GOTO 2070
2050 IF B%=35% THEN 2070
2060 B%=B%-2 : X%=X%-2
2070 IF A%<5% THEN A%=5%
2080 IF A%>61% THEN A%=61%
2090 IF B%<5% THEN B%=6%
2100 IF B%>64% THEN B%=64%
2110 GOTO 730
2120 IF S2%=3000% THEN 1110
2130 IF INP(56)=80 THEN 30
2140 IF DOT(36,40) THEN 1370 ELSE 490
2150 END
```

Szerkesszük együtt!

A MAGAZIN Játékprogramok rovata sokszor szűknek bizonyul a beérkező programokhoz képest. Ezért azt tervezzük, hogy a legelterjedtebb gépekre (úgyis sejtik, hogy mikre) egy különkiadványt állítunk össze. Csupa játékprogram – mondjuk ez lesz a címe.

A megjelentetésre szánt programokat kérjük november 1-ig küldjék be. A fő szempont az, hogy a játék minél jobb legyen, de ne feledkezzenek el a leendő olvasókról sem. Ők nem a játékkal, hanem a beírással kezdik, tehát ne legyen terjengős a program. Arról se feledkezzenek meg, hogy az olvasók érteni is akarják a listát, tehát néhány REM-et megérdemelnek. Ha gépi kódú rutin is van a programjukban, akkor szánjanak néhány szót a bevitelre is. A programokat kazettán kérjük.

Ha elég szívósak és kitartóak, nem rettennek vissza sem a programozástól, sem a hibakereséstől, és ha megbízhatóak és precízek, akkor várjuk jelentkezésüket *október 15-ig*, hogy együtt szerkesszünk.

Az olvasó írja

Az olvasó írja tanulságos leveleit, mi pedig olvassuk. Miután sok levél érkezett, egy részüket az illetékes rovatszerkesztőkhöz továbbítottam, akik vagy személyesen, vagy a rovatban válaszolnak. Most azokból a levelekből idézek, amelyek – véleményem szerint – különösen érdekelhetik olvasóinkat.

Hajdú Csaba, Budapest,

Bocskai út 34/a. 1113

Köztudott, hogy nálunk (és valószínűleg a világon mindenütt) mindenki másolással szerzi be programjai nagy részét, ha nem az összeset. Másolóprogramok is tucatszám akadnak, bár elvben csak saját programokat szabad másolni. Ezt ugyan senki nem veszi komolyan, de nem tudom, hogy jogi szempontból célszerű-e a ZX81 jelfrissítőt reklámozni, vagy ilyesmit írni, hogy „amellyel az elvben másolhatatlan gyári programokat is reprodukálni lehet". Arról nem is beszélve, hogy elvben másolhatatlan program nincs, legfeljebb a gyártók a gyakorlati másolást igyekeznek megnehezíteni.

Igaza van.

Nagy Péter, Budapest,

Toldy F. u. 72/b. 1015

Megkaptuk a kitöltött kérdőívet és hozzá némi indoklást.

Nos hát! Minden nemleges válaszomat indoklom.

1. A rovatszerkezet. A cikkek terjedelmének aránya szerintem egy kicsit el van tolódva rossz irányba. Köznapi nyelven ezt úgy mondanák, hogy sok a „söder". Egyes közérdekű cikkek túl rövidre sikerülnek (például: Piac), és egyes, szűk rétegeket érintő cikkek pedig túl terjedelmesek (APL programnyelv).

2. Programozástechnika. Lásd 1.

3. Ember – gép kapcsolat. Szerintem felesleges ezt a témát egy teljes oldalon át fejtegetni, hiszen a μ M-t úgyis csak az olvassa, aki ismeri a problémákat és munkálkodik a „jó ügy" győzelméért.

4. Az olvasó írja. Azt hittem, hogy ez a jelenség el fog múlni, de nem. Szerintem felesleges mindenkinek a gratulációját leírni a lapba, mert általános vélemény, hogy a μ M nagyon jól sikerült dolog.

5. Játékprogramok. Im, előjött a vesszőparipám! Sok más számítógépes újságot olvastam, de mindenhol több volt a szoftver. Szerintem egy ilyen témájú újság jelentős részét a programoknak kellene kitenni.

6. Rövid és ravasz programok. Kevés! Különben GOTO 5.

7. Ez a hét könyv nem egészen új már, és az olvasók nagy része már ismeri és *utálja* őket.

8. Sakk. Erre a témára kár ennyi helyet áldozni. A laikus nem tud sakkprogramot írni (géphiány, nehéz algoritmus, a gép minősége), aki pedig tud, az előbb-utóbb rátalál erre a problémára.

Az ízlések és pofonok különbözők. Ami a 4. pontot illeti, egyetértek, már nem is tesszük, annak ellenére, hogy sok dicséretet kapunk. Eltesszük magunknak. A könyvekkel kapcsolatban nekem és néhányunknak más a véleménye.

Dr. Börcsök Sándor, Szeged,

Batthyány u. 33. 6722

Minden totem figuránál többet ér, ha nem emelik az árat és nem csökkentik a tartalmat hirdetéssel.

Az árat nem a szerkesztőség állapítja meg; mi oroszlánként küzdöttünk, hogy az ár ne változzék. Megváltozott. Hirdetések nélkül ma nem lehet lapot kiadni. A százezer alatti példányszám, valamint a rendkívül magas előállítási és postai költség miatt hirdetések nélkül a lapot nem tudnánk még ezen az árszinten sem tartani.

Ladányi Gábor, Bükkszentkereszt,

Béke u. 24. 3557

Nagyon jónak találom valamennyi rovatukat. Különösen jónak a számítógép hardverjéről, belső felépítéséről, a számítógépek kiegészítéséről szóló rovatukat.

Én kevésnek találom az ilyen számítógépépítő, számítógép-kiegészítő kapcsolásokat. Nagyon jó lenne, ha párhuzamosan két vagy több számítógép építésének leírását közölnék, amatőr és kezdő szinten. A számítógép szélesebb felhasználását elősegítő, kiegészítő kellékek közreadása is lehetne több. Az építést segítené, ha az alkatrészek árát, beszerzési helyét is megadnák.

És még egy fontos „szépséghibára" szeretném felhívni figyelmüket: a rajzjelek nem voltak szabványosak, és ezért nagyon nehezen igazodtam el a rajzokon. Az alkatrészek értéke is pontatlan volt, különösen a kondenzátoroknál. Az IC-k sem voltak egységes jelölésűek. Nem tudtam kiigazodni, a tápáram negatív, vagy pozitív részére kössem-e a nyíllal ellátott 5 V-ot. A szabványos rajzjeleket, jelöléseket különben Magyari Béla: „Rajzjelek és jelölések az elektronikában" című könyvéből ismerem. A könyv a Műszaki Könyvkiadó gondozásában jelent meg 1980-ban.

Mire ez a szám megjelenik, remélem, olvasónk kívánsága teljesült. Az egységes jelölésre vonatkozóan valamit tenni fogunk.

Nagy Lajos és Sárai Ferenc, Komárom,

Gépipari Szakközépiskola, Csehszlovákia

Első külföldi levelezőinket külön is üdvözlöm. Hosszú levelükből néhány részlet (a dicséretet nem volt szívem kihagyni!):

Lapjukat nagy lelkesedéssel fogadtuk, és nagy örömet is szerzett nekünk, ugyanis mínálunk ilyen, vagy ehhez hasonló újság nem létezik, és egyhamar nem is várható a megjelenése. Szerencsénkre idáig még mind a két kiadást meg tudtuk szerezni, amelyet egymás kezéből kapkodva olvasunk el és áttanulmányozzuk, ha már egyszer el is olvastuk, ugyanis sok érdekességet tudtunk meg idáig is belőle.

Éppen most hoztunk rendbe egy ZX81-est (billentyűzet probléma), amely köré szeretnénk egy kis klubot szervezni. Sajnos még most nem tudtunk programokat küldeni, de reméljük, a közeljövőben lesz rá mód. Szeretnénk még megkérdezni, hogy miképp lehetnénk a HCC levelező tagjai.

Válaszlevelet küldtem.

Ágost Zoltán László, Szolnok,

Orosz Gy. út 38. 5000

Munkakörülményeim változása miatt már jó néhány éve csodálattal figyelem a változásokat – bár közvetlen kapcsolatom nincs a számítástechnikával. Éppen ezért kérem Önöket, hogy lapjukban az ilyen egészen (újra)kezdőknek is engedjenek néhány oldalt.

Ezen túl: hamarosan négy gyerekem lesz, mindenesetre három már van. Szeretném, ha számukra a számítógép kezelése már természetes lenne. Ezért kérem, hassanak az illetékes társadalmi szervekre, hogy ne akadályozzák, hanem inkább segítsék a külföldi számítógépek behozatalát – ez biztos, hogy nemcsak az egyént, de az országot is gazdagítaná. Számomra szörnyű, hogy a legolcsóbb számítógépet (a Primóra gondolok) is csak több éves gyűjtéssel szerezhetem meg. Ezen a téren nem lehetne előrelépni?

A lapot az (újra)kezdőknek is szánjuk. Mire ez a szám az olvasóhoz kerül, talán lesz már olcsó Kit-számítógép. Optimista vagyok.

Gera János, Csanytelek,

Ady Endre u. 3. 6647

Ebben a rovatban felfigyeltem Raffai Zsolt kecskeméti diáktársam levelére, amelyben hiányolja a HT–1080Z típusú számítógéphez a pontos hangskálakódokat. Ez a kérdés engem is régóta foglalkoztat, viszont nekem még csak nem is sikerült megtudnom, hogy honnan szerezhetném be ezeket az adatokat. Így a rendelkezésemre álló 2–3 kód segítségével próbáltam meghatározni a továbbiakat is, valamiféle logika segítségével. A kapott adatokat programba ültetve, megpróbáltam pontosítani a hallásom alapján. Az így kapott kódok pontosságában azonban még most sem vagyok biztos, ezért az Önök segítségét kérném: ha lehet, küldjék el számomra is a hangskála kódjait.

Amennyiben valamiféle azonosságot találok az általam számított és a hiteles adatok között, és ha ezzel a számolgatással segítséget nyújthatok a további érdeklődőknek, akkor a számolgatásom menetét levélben szívesen megírnám.

Levelét a Híradástechnika Szövetkezetnek megküldtem. Számolgatásának menetét cikk formájában küldje el, közöljük.

Voznek Csaba, Zirc,

Reguly Antal u. 63. 8420

Elsőéves villanyszerelő vagyok, a 306. I. SZ. Intézetben tanulok. Év elején, amikor az iskolánkba megérkezett két HT–1080Z számítógép, majdnem az egész osztály járt a szakkörbe. Sajnos mostanra már csak négyen maradtunk. Mind a négyen elég jól tudunk programozni. Nagy örömmel fogadtuk a μ M megjelenését. Szerintem nagy segítséget nyújt a

kezdőknek és a haladóknak is. De konkrétabban kellene érintenie az egyes géptípusok programozását.

Egy számítógép megvásárlása nem az én pénztárcámhoz van mérve, sőt még egy számítógép megépítéséhez sincsenek meg a feltételeim. Így hát kénytelen vagyok a klubokra és az iskolára hagyatkozni.

Azt remélem, hogy az ilyen mértékű érdeklődéscsökkenés a nem jellemző, ritka kivételek közé tartozik. Ezért közöljük.

Jáhni Csaba tanuló, Szeged,

Székely sor 15/B. 6726

1. Növeljék meg a lap példányszámát, hogy minden érdeklődő hozzájusson! (Ez gondolom a problémák ellenére is jó bevétel lenne.)

2. Helyezzenek el az újságban megrendelőlapot!

3. Alakítsanak egy csoportot, amely számítástechnikai cikkekkel látná el a magánszemélyeket. Ez úgy működhetne, hogy valutában kellene fizetni, és előre rendelni. Az összegyűlt rendeléseket havonta lehetne behozni az országba, és kb. 5 százalék felár fedezné is a vele járó költségeket. Még az sem lenne túl nagy gond, ha a vámot is valutában kellene fizetni.

4. Ki lehetne írni egy új, határidő nélküli pályázatot személyi számítógépek és perifériák tervezésére. A feltételek az egyszerűség, olcsóság, dollár nélküli alkatrészek, több programnyelv ismerete stb. lehetnének. A jól sikerült terveket magánvállalkozók vagy téesz melléküzemágak gyárthatnák.

Javaslatait változtatás nélkül közöljük; tulajdonképpen el lehet gondolkozni rajtuk. Az utóbbit meg is tettük, Kit-számítógépre kértünk javaslatot. Kaptunk is, csak az árban nem jutottunk közös nevezőre. A lap példányszámát 15 ezerről 35 ezerre növeltük. Reméljük, hogy ez sem elég!

Nagy Jenő ny. tanár, Hatvan,

Mező Imre u. 7. 3000

Elsősorban nagyon érdekel a sakkprogramozás. Végső célom nem a játék programozása, ez csak közbülső cél. Egy ismeretelméleti és tudományelméleti kérdés izgat. A kölcsönhatásos rendszerek megértése, és mint számítógép-rendszerek gépi szimulációja, ami izgat. Komplex rendszeren értem azt a rendszert, ahol a részek összege nem egyenlő az egésszel. Ezen plusz kimutatása a gép révén, amelyet nem kötnek a plauzibilis szemléleti formák. Komplex jellegű sakktanulmányok programozása is érdekel. A komplex rendszerek szimulációjához akarok eljutni – természetesen számítógép segítségével.

Várjuk a sakkprogramozásban elért eredményeit. Sok sikert kívánunk.

Perlaki Károly, Budapest,

Liliom u. 54. 1094

A legutóbbi számból hiányoltam a Commodore 64 és VIC-20-as gépekre írt (játék)programokat; csak a ZX81-gyel foglalkoztak. Köztudott, hogy a ZX81 a maga csekély 1 k RAM-jával nem tartozik a legjobb gépek közé, a ZX-Spectrum viszont már túl drága. Így esett a választásom a VIC–20-ra, bár ennek sincs túl sok RAM-ja.

3583 bájttal csak rövidebb programok futtathatók. Az lenne a kérdésem, milyen költséggel lehetne a VIC–20 memóriáját kiterjeszteni legalább 48 k RAM-ra, hogy segítségével például egy sakkprogram megírható legyen. Nem vagyok (még) a téma kiváló szakértője, így Önökhöz, a szakemberekhez fordulok.

Tagja vagyok a Magyar Sakkszövetség Sakkszámítógép Bizottságának, amelynek dr. Lindner László az elnöke. Úgy tudom, most fog létrejönni bizonyos szervezeti együttműködés a Sakkszövetség és a Társaság között.

Sakkszámítógépekkel egyébként már 4 éve foglalkozom, segítségükkel levelezési sakkjátszmáimat elemzem. Nagyon jó szolgálatot tesz a gép a bonyolult állásokban való eligazodásban és a legjobb lépés megtalálásában. Segítségével már nagyon sok játszmát nyertem meg, illetve mentettem döntetlenre. (Itt meg kell jegyeznem, hogy a gép segítségével csupán a saját lépéseimet ellenőrzöm, nem hagyatkozom szolgaian a gép által megadott lépésekre.)

Kaptunk néhány levelet, amelyben a sakkprogramozást haszontalannak ítélik. Köszönöm a cáfolatot. Egyébként nemcsak ZX81 programokat közlünk, ezt bizonyára tapasztalta. Szeretnénk az országban fellelhető valamennyi gépre programokat ajánlani. A VIC–20 bővítésével kapcsolatban levelét a HCC-nek továbbítottam.

Dolongorszky József,

Miskolc-Perecses

A Miskolc-Perecesen működő 104. I. SZ. Intézet 3. osztályába, háztartásigép-szerelő szakra járok. Az Önök lapjából szereztem tudomást egy számomra új és érdekes kezdeményezésükről, az NJSZT HCC-ről. Ez a tény, hogy már ilyen „amatőr" klub is működik hazánkban, nagyon felkeltette érdeklődésemet. Viszonylag régóta foglalkozom elektronikával, és amikor először farkasszemet néztem magával a számítógéppel, műszaki érdeklődésem éles ívű kanyarral a számítástechnika felé fordult. Minél jobban belemerültem a számítástechnikába, annál inkább a gép teste érdekelt. És mivel nem engedélyezték, hogy a nehezen szerzett HT-nket alkotóelemeire szedjük szét, tudásszomjamat a szakkönyvekből kezdtem kielégíteni. De hardver témával foglalkozó könyvet alig lehet találni, és így e hír hallatára ugyancsak felélénkültem. Véleményem szerint egy gép megépítésével még közelebb kerülhetek a számítógép titkainak megértéséhez.

... és mi lesz a háztartási gépekkel?

Török József, Pápa,

Anna tér 3/a. 8500

A pápai Mezőgazdasági és Élelmiszeripari Szakközépiskola Szakmunkásképző Intézetében húsfeldolgozónak tanulok, és most végzek. De ebben az iskolában folytatom tanulmányaimat, így további négy évig lehetek az iskolagép „birtokosa".

Számítástechnikával egy hónapja foglalkozom, és annyira megtetszett, hogy elhatároztam, hogy amit lehet, megtanulok róla. Egyik barátomtól kaptam kölcsön az Önök magazinját, és bevallom, ez az első igazán jól szerkesztett, érdekes témákkal foglalkozó lap, ami a kezembe került.

A számítástechnika már a húsiparból is csábít!

Szeklencei Tamás, Budapest,

Váci út 134/B. 1138

Elemezgettem az idei szám cikkeit. Hogy is mondjam? Nem tetszett igazán minden. Mindjárt az elején kezdve: nagyon érdekes volt a Kozma Lászlóra emlékező cikk. Nemcsak a matematikust ismerhettük meg, hanem a számítógépek történetéből is egy fejezetet. Örömmel olvastam, hogy egyik olvasótársam cikksorozatot fog közölni a számítógépek „történelméről".

Igazán segíti az elméleti „alapozást" a Tanfolyam. Egyszerű, jól érthető. Ugye folytatják?

Mi az, amit az olvasók gyakorlatban használhattak, hasznosíthattak ebből a számból (az előzőeken kívül)? Az „A/D átalakító TRS80 Model 1-hez", a „Magnetofonilleszto VIC–20-hoz", az első „ZX81-tulajdonosok, figyelem" című cikket, valamint a játékprogramokat. Voltak olyan cikkek, amelyeket kimondottan feleslegesnek éreztem. Nem tetszett a jelbeszéd; hosszú elmélkedés közismert kérdésekről, s mindez anélkül, hogy bármilyen információt is közölne.

Hosszú leveléből csak ennyit idézünk. Nem tetszett olvasónknak az APL leírás és főleg a hirdetések. Ez utóbbiról már sokszor írtam. Ami a levelezési rovatot illeti, egyre több ilyen tartalmas levelet kapunk, így egyre könnyebb ezt a rovatot is színvonalasabban szerkeszteni.

Hegedüs Balázs, 3. oszt. ált. isk. tanuló,

Budapest, Síp u. 6. 1075

ZX81 gépem van. Több legyen a játékprogram, és jó lenne, ha olcsóbb lenne a felszerelés.

Azt hiszem, hogy legifjabb olvasónk véleményét közöljük; az utóbbi megjegyzésével teljes mértékben egyetértek.

Szilágyi Gábor, Debrecen,

Micsurin u. 86. 4027

Lakásköltözés miatt fiam, lányom más iskolába került. A fiam kimondottan megutálta az iskolát, a nem tanulása már tragédia volt. A számítógép kezdi meghozni a tanulási kedvét; olyan programokat vettem fel (és csináltam is), mint például: „A legnagyobb közös osztó" programcikkben megjelent. Azt hiszem, az én esetem nem egyedi, más szülőknek is lehet ilyen segítség, ezért nemcsak matematikai, hanem más tantárgyi program is jöhetne.

Ennek mi is örülünk, ha kapunk más oktatóprogramokat, közöljük.

Valamennyi levelezőnk tanácsait, dicséretét és kritikáját ismételten köszöni:

KOVÁCS GYŐZŐ

ISMERJE MEG ÉS VEGYE IGÉNYBE AZ

OKISZ Szervezési és Számítástechnikai Vállalat

SZOLGÁLTATÁSAIT

ÜGYVITEL-SZERVEZÉS ÉS GÉPESÍTÉS

A ROBOTRON 1372, 1355 és 1711 típusú gépekre kidolgozott és nagy számban telepített rendszereink:
- anyagkönyvelés,
- költségvetés készítés
- számlázás
- fogyóeszköz könyvelés
- főkönyvi és folyószámla könyvelés
- bérszámfejtés, bérügyvitel
- készárukönyvelés
- gyártáselőkészítés
- utókalkuláció.

A több száz alkalmazói rendszerből a feladatoknak, az igényeknek megfelelő referencia helyet tudunk bemutatni.
Az ügyvitelgépesítésnél „kulcsrakész" feladat ellátását biztosítjuk, mely kiterjed a gépek beszerzésére, a segédanyagok biztosítására, oktatásra és betanításra is.

Ügyvitelszervezési Osztály
Telefon: 336-399

MIKRO-SZÁMÍTÓGÉPES ALKALMAZÁSOK

A VIDEOTON VT 20 és VT 30, a Robotron 5100-as gépcsalád és FLOPPYMAT SP mikroszámítógépek alkalmazására kidolgozott típusrendszereink lehetőséget adnak a fokozatos alkalmazásfejlesztés megvalósítására:
- anyag-, áruforgalmi rendszer,
- készletgazdálkodási rendszer,
- termelést előkészítő és támogató szolgáltatások,
- kereskedelmi információs rendszer.

Szervezési Főosztály
Tel.: 426-936

SZÁMÍTÓGÉPES FELDOLGOZÁSOK

Saját fejlesztésű rendszerek teljes körű feldolgozásait biztosítjuk kötegelt üzemmódban, adatrögzítéssel és adatkonvertálással.
A feldolgozások országos alkalmazási lehetőségét a SZÜV megyei számítóközpontjaival együttműködve valósítjuk meg. Vállalkozunk egyedi feldolgozási rendszerek kifejlesztésére, mintarendszereink adaptálására, applikálására.
- anyaggazdálkodási általános rendszer,
- építőipari anyaggazdálkodási rendszer,
- termelésirányítást támogató rendszer,
- ipari szövetkezeti irányítási rendszer,
- áruforgalmi rendszer.

Számító központ
Telefon: 210-808

OKISZ SZERVEZÉSI ÉS SZÁMÍTÁSTECHNIKAI VÁLLALAT Budapest 8. Pf. 247. 1445 Telefon: 340-102

A mesterlövész nem biztos, hogy jó katona. A jó katona azonban biztosan alaposan ismeri a harci eszközeit. A programozás eszközeinek virtuóz kezelői csak ritkán jó programozók. A jó programozó azonban biztosan alapos ismerője a programozási eszközöknek. És nem minden jó programozó jó számítástechnikus is. De olyan jó számítástechnikus, aki rossz programozó volna, olyan nincs. Rovatunk célja az, hogy sajátos eszközeivel elősegítse, hogy lapunk minél több jó számítástechnikus képzéséhez járuljon hozzá azáltal, hogy segíti őket a jó programozóvá válásban és a nagyon gazdag szakmai fegyvertár, harcászat és hadászat megismerésében.

Pedagógiai elvünk a „profi" gyakorlati alkalmazásra nevelés. Ezért tudatosan kerüljük a naiv, nem komoly témákat, de – ott, ahol ez helyénvaló – törekszünk a könnyed, esetleg játékos megfogalmazásra, vállalva az ezzel járó többletmunkát is, mert hiszen egy komoly téma játékos, szemléletes de szigorúan szakszerű tálalása művészet. Művészet, mégpedig a legmagasabbrendű – nemcsak előadó, hanem alkotó – művészetek közül való.

A megbízható programozási tudáshoz rengeteg gyakorlat szükséges. Ahhoz pedig nem csekély idő kell, amíg minden fontos területről elég sok feladat megoldása révén kialakul a megfelelő jártasság és készség.

Rovatunk fő célja kielégíteni azoknak a jogos igényét, akik rendezett és válogatott feladatsorozatok végigdolgozása révén gyorsabban és könnyebben kívánnak szakmai gyakorlottsághoz jutni, nem kivárva míg az élet megfelelő mennyiségű feladatot hoz elébük. Rovatunk számítástechnikai kiegészítő edzésként is felfogható, hasonlóan az ökölvívók erősítő, kitartást fokozó, ellenállóképességgyarapító favágásához. Ehhez a favágáshoz – mondhatnánk helyette bitaprítást is – azonban nem kell sem fa, sem fejsze, csak sok-sok feladat és céltudatos szorgalom. És van még egy különbség. A fejsze éle, minél több fát vágnak vele, annál tompább lesz. Az elme azonban élesedik e favágástól.

A feladatok egy megoldását mindig, még ugyanabban a lapszámban közöljük, de csak a pillanatnyi elakadás megszüntetése és a végeredmény ellenőrzése érdekében forduljunk ezekhez.

Jó favágást, sikeres bitaprítást kívánunk.

If nélküli megoldások

A következő feladatsorozatok egyes gyakori függvények teljesítőképességének megismerésére és a használatukra vonatkozó készség kialakítására szolgálnak.

Készítsünk programokat if utasítás használata nélkül, a következő függvények helyettesítési értékének kiszámítására! (A több lehetséges megoldásból legalább egyet, néha azonban többet is közlünk, de csak a függvény értékének kiszámítását vezérlő programsort.)

Azok számára, akik esetleg nem ismerik, közöljük, hogy a sgn az ún. „signum" függvény jele, amely pozitív értékekhez a +1, negatívakhoz a −1 értéket, zérushoz pedig a zérust rendeli. Az abs pedig az abszolútérték függvény jele, amely negatív értékhez annak −1-szeresét, nem negatív értékhez pedig annak +1-szeresét rendeli.

A közölt megoldások természetesen csak ideális, valós műveletek használatánál adnak minden esetben helyes eredményt. Gépi műveletekkel dolgozva, az egyes szakaszok határának közelében számítani kell pontatlanságokra.

•

A közölt feladatok a rovatszerkesztő „Elemi számítástechnikai példák és feladatok" c. gyűjteményéből valók. A megoldásokat Takács Ferenc ellenőrizte.

```
JELMAGYARAZAT,
MEGJEGYZES

^  HATVANYOZAS
<= KISEBB VAGY EGYENLO
>= NAGYOBB VAGY EGYENLO
<> NEM EGYENLO
A 9.-11. FELADATBAN A<B,
A 12. FELADATBAN  A<=B.

FELADATOK

1. Y=-1,    HA     X < 0;
   Y= 0,    HA     X = 0;
   Y= 1,    HA     X > 0.

2. Y= X,    HA     X >=0;
   Y=-X,    HA     X < 0.

3. Y=X/ABS(X),HA X <>0;
   Y= 0,        HA X = 0.

4. Y= 0,    HA     X < A;
   Y= 1,    HA     X = A;
   Y= 0,    HA     X > A.

5. Y= 0,    HA     X <=A;
   Y= 1,    HA     X > A.

6. Y= 0,    HA     X < A;
   Y= 1,    HA     X >=A.

7. Y= 0,    HA     X >=A;
   Y= 1,    HA     X < A.

8. Y= 0,    HA     X > A;
   Y= 1,    HA     X <=A.

9. Y= 0,    HA     X <=A;
   Y= 1,    HA     A<X<B;
   Y= 0,    HA     B <=X.

10. Y= 0,   HA     X <=A;
    Y= 1,   HA     A<X<=B;
    Y= 0,   HA     B < X.

11. Y= 0,   HA     X < A;
    Y= 1,   HA     A<=X<B;
    Y= 0,   HA     B <=X.

12. Y= 0,   HA     X < A;
    Y= 1,   HA     A<=X<=B;
    Y= 0,   HA     B < X.

13. U=-(Y-P), HA   X < 0;
    U= Y-P,   HA   X >=0.
    (EZ PAROS FUGGVENYBOL
    PARATLAN KEPZESERE IS
    AD EGY LEHETOSEGET.
    UGYANIS HA Y=F(X) ES
    Y AZ X-NEK PAROS FUGG-
    VENYE, ES P=F(0),AKKOR
    U PARATLAN FUGGVENYE
    LESZ X-NEK.)

14. U=-Y+P,  HA    X < 0;
    U= Y+P,  HA    X >=0.
    (EZ PARATLAN FUGG-
    VENYBOL PAROS KEPZESE-
    RE IS AD EGY LEHETOSE-
    GET. UGYANIS HA Y=F(X)
    ES Y PARATLAN FUGGVE-
    NYE X-NEK, ES P TET-
    SZOLEGES ALLANDO,
    AKKOR U PAROS FUGGVE-
    NYE LESZ X-NEK.)
```

MEGOLDÁSOK

A „Favágás" feladatahoz

```
1. LET Y=SGN(X)

2. LET Y=ABS(X) VAGY
   LET Y=SGN(X)*X

3. LET Y=SGN(X)

4. LET Y=1-ABS(SGN(X-A)) VAGY
   LET Y=1-SGN(ABS(X-A))

5. LET Y=1+SGN(SGN(X-A)-1) VAGY
   LET Y=(SGN(X-A)+ABS(SGN(X-A)))/2

6. LET Y=SGN(SGN(X-A)+1)

7. LET Y=1-SGN(SGN(X-A)+1) VAGY
   LET Y=(SGN(A-X)+ABS(SGN(A-X)))/2

8. LET Y=SGN(SGN(A-X)+1) VAGY
   LET Y=-SGN(SGN(X-A)-1)

9. LET Y=1+SGN(SGN((A-X)*(X-B))-1) VAGY
   LET Y=1-SGN(SGN((X-A)*(X-B))+1) VAGY
   LET Y=(1+SGN(SGN(X-A)-1))*(1-SGN(SGN(X-B)+1))

10. LET Y=SGN(SGN(B-X)+1)*(1+SGN(SGN(X-A)-1))

11. LET Y=SGN(SGN(X-A)+1)*(1-SGN(SGN(X-B)+1))

12. LET Y=SGN(SGN((A-X)*(X-B))+1) VAGY
    LET Y=SGN(SGN(X-A)+1)*SGN(SGN(B-X)+1)

13. LET U=SGN(X)*(Y-P)

14. LET U=SGN(X)*Y+P
```

Az „Agyafúrmány" előző számban közölt feladataihoz

1. feladat:

```
10 REM A PROGRAM A
20 REM NYUGDIJJARULEK
30 REM SZAMITASAT VEGZI
40 FOR I=0 TO 10
50 READ A(I): NEXT I
60 INPUT"FIZETESE :";B
70 J=3: REM KEZDO X-EK
80 FOR I=0 TO 10
90 IF B)=A(I) THEN J=J+1
100 NEXT I: IF J)9 THEN J=J+1
110 C=B/100*J: E=C-INT(C)
120 IF E)=0.5 THEN C=C+1
130 C=INT(C): D=B-C
140 PRINT " FIZETESE  :";B
150 PRINT " LEVONAS   :";C
160 PRINT "KIFIZETENDO:";D
170 GOTO  60: REM ISMETLES
180 REM FIZETESI HATAROK
190 DATA 2101,2601,3301,4301
200 DATA 5301,6301,7301,8301
210 DATA 10301,12301,14301
```

2. feladat:

```
10 REM OSZTOK SZAMANAK SZAMITASA
20 DIM B(899): M=20
30 FOR I=2 TO 499
40 A=INT(100/I)*I
50 IF A(100 THEN A=A-I
55 PRINT I;A
60 FOR J=A TO 999 STEP I
70 B(J-100)=B(J-100)+1: NEXT J
80 NEXT I: FOR K=100 TO 999
90 IF A)B(K-100) GOTO 140
100 IF M=B(K-100) THEN N(S+1)=K
110 IF M=B(K-100) GOTO 140
120 M=B(K-100): FOR L=0 TO 10
130 N(L)=0: NEXT L: S=1: N(0)=K
140 NEXT K: REM NYOMTATAS
150 PRINT "OSZTOK SZAMA  A SZAMOK..."
160 PRINT M;N(0);N(1);N(2);N(3)
170 PRINT " *** V E G E *** "
180 STOP: END
190 REM ************************
200 REM OSZTOK SZAMA   A SZAMOK
210 REM
220 REM       30          840
230 REM
240 REM ************************
```

3. feladat:

```
10 REM SZAMJEGY MEGHATAROZASA
15 I=0
20 INPUT "KEREM A SZAMLALOT";K
30 INPUT "KEREM A NEVEZOT";M
40 INPUT "HANYADIK JEGYET KERI";L
50 IF K)=M GOTO 90
60 K=10*K: I=I+1: IF L)I GOTO 50
70 PRINT "A SZAMJEGY ERTEKE :  0"
80 GOTO 15: REM FOLYTATAS
90 K=K-M: J=J+1: IF K)=M GOTO 90
100 I=I+1: IF L)I GOTO 130
110 PRINT " A SZAMJEGY ERTEKE:";J
120 GOTO 15: REM FOLYTATAS
130 J=0: GOTO 60: REM J NULLAZASA
```

4. feladat:

```
10 REM VELETLENSZAM
20 A=RND(999): B=RND(999)
22 REM NEM MINDIG KELL!!!
25 A=INT(1000*A): B=INT(1000*B)
27 REM AZ ELOZO SOR!!
30 IF A=B GOTO 20: REM *
40 PRINT "A SZAMOK:";A;B
50 PRINT "GONDOLJ EGY"
60 PRINT "EGESZ SZAMOT!"
70 PRINT "AZ OSZTAS UTANI"
80 INPUT "MARADEKOK:";C,D
90 A=A-C: B=B-D
100 REM KOZOS OSZTO
110 FOR I=2 TO 499
120 E=INT(A/I)*I
125 PRINT E;A:FOR P=0 TO 999:NEXT P
130 IF E(>A GOTO 170
140 M=INT(B/I)*I
150 IF M(>B GOTO 170
160 PRINT "AZ OSZTO:";I
170 NEXT I: PRINT "V E G E"
180 GOTO 10: REM FOLYTATAS
```

Mikro-számítógépek és rádió-amatőrök

A több, mint fél évszázados nemzetközi rádióamatőr-mozgalom hatalmas tömegeket mozgat meg a világon. Tagjai szabad idejüket feláldozva fejlesztik készülékeiket, keresnek összeköttetést távoli, több ezer kilométerre lévő partnereikkel. Ezek a kísérletek mindig is együtt haladtak az adott kor csúcstechnológiájával, esetenként egy-egy lépéssel meg is előzve azt. Ennek köszönhetően a gyakorlatban ma már megtalálhatók a kontinensek közötti színes képátvitel, az űrtávközlés és a digitális adatátvitel szükséges eszközei is. Ez utóbbi hagyományosan a postai telexforgalomból ismert mechanikus távgépírókat használja. A zörgő, nehezen beszerezhető és mai szemmel nézve lassú és nagyméretű gépek azonban akadályai voltak a széles körű elterjedésnek.

Az előrelépést itt is, mint az élet sok más területén, a mikroprocesszorok megjelenése tette lehetővé. Felhasználásukkal a régi gépek lecserélhetők voltak olyan terminálokra, amelyek olcsón beszerezhetők és számos kényelmi szolgáltatást is nyújtanak – például szövegszerkesztést. Mindezek ellenére ezek mégis csak „telexgép"-nek tekinthetők, hiszen megmaradt az ötbites kód és a 45 Baud adatátviteli sebesség, amely másodpercenként mintegy hét karakter átvitelét biztosítja.

A személyi számítógépek tömeges elterjedése megvetette az alapját az ennél korszerűbb információátviteli módok alkalmazásának. Megjelentek a gyorsabb, hibavédett átvitelt megvalósító eljárások. A legelőremutatóbb, forradalmi változást azonban a rádióamatőr rendeltetésű számítógép-hálózatok kialakítására tett kezdeti lépések jelentik.

Információs rendszerek

A rádióamatőrködés jellegéből adódóan társas hobbi. A vele foglalkozók gyakran cserélik ki ismereteiket, információikat egymással. Ez azonban esetlegessége miatt nem az igazi, hiszen a vonal túlsó oldalán a partnernek személyesen kell jelen lennie.

Iktassunk be a láncba egy mikroszámítógépet, és kapcsoljuk össze egy rádió adó-vevővel. Ezután az információk nagy távolságból, a nap bármely szakában elérhetők lesznek. Egy ilyen rendszer ráadásul mindkét irányban biztosítja az átvitelt, tehát az elektronikus újságot maguk a felhasználók szerkeszthetik.

Az ezt a célt szolgáló hálózatok természetesen nemcsak rádión, de telefonon is üzemeltethetők, mint arra számos példát találhatunk a fejlett számítástechnikai kultúrájú országokban. Ezek többnyire egyes számítógépklubok tagjait kötik össze, illetve ré-

```
hg5bme de hg0dy .rita

- here is: rita -
 welcome at hg5bme

 uj infok:
 news 34
 cndx
 intr 11
 news 32
 satl 1
 diar 1
```

1. ábra. Belépés

szükre szolgáltatnak közérdekű információkat.

A RADIR

Felismerve az előzőekben elmondottakat, a Budapesti Műszaki Egyetem Rádióklubja 1982-ben hozzáfogott egy Rádióamatőr Információs Rendszer, a RADIR létrehozásához. Ez a tervek szerint egy, az ország nagy részére kiterjedő hálózat, amelyben az egyes felhasználókat erősítőállomások kapcsolják össze, és amelyben több, az információk tárolására és előállítására szolgáló számítógép található. Ezen keresztül tetszőleges, rádióamatőr jellegű adat, hír cserélhető és továbbítható. A RADIR első elemei már a gyakorlatban vizsgáz-

```
hg5bme de hg0dy
 .read news 31

news 31              1820 ut 28.03.84     272     read

 a francia radioamator szovetseg f8ref  hivojelu kozponti
 allomasa  rtty  informacios bulletint sugaroz a kovetkezo
 idopontokban:

   minden hetfon    2000 ut  kezdettel  3.590 (+-5) khz-en
   minden szombat   0700 ut  kezdettel  7.038 (+-2) khz-en

 ...info radio-ref

1505 ut 19,06.84  next?
```

2. ábra

```
hg5bme de hg0dy
 .dir

     filename          created          size   status  ky   4
intr 14          1742 ut 27.03.84      456     read
news 31          1820 ut 28.03.84      272     read
refr 1           1608 ut 31.03.84.     156     read
refr 5           1619 ut 31.03.84.     425     read
refr             1619 ut 31.03.84.     366     read
ucsw 1           1623 ut 31.03.84.     424     read
ucsw 2           1625 ut 31.03.84.     261     read
ucsw 4           1627 ut 31.03.84.     322     read
ucsw             1629 ut 31.03.84.     197     read
news 29          1130 ut 05.04.84      1730    read
adrs             1614 ut 11.05.84.     258     read
cndx 3           1137 ut 23.05.84.     354     read
news 33          1107 ut 28.05.84.     406     read
hg2rj            0636 ut 05.06.84      157     private
hg6vv            1740 ut 05.06.84      341     read
intr 11          1645 ut 09.06.84      1099    read
diar 1           1722 ut 09.06.84      1029    read
cndx 8           0730 ut 10.06.84      1145    read
news 34          1343 ut 11.06.84      378     read
chng qtc 3       0553 ut 12.06.84      88      open
satl 3           1424 ut 13.06.84      519     read
satl 4           1439 ut 13.06.84      652     read
satl 1           1743 ut 13.06.84      506     read
satl             1748 ut 13.06.84      247     read
meghivo          1018 ut 14.06.84      589     open
cndx 20          0645 ut 16.06.84.     530     read
cndx 21          1041 ut 16.06.84.     238     read
cndx 22          1309 ut 16.06.84.     387     read
cndx 23          0453 ut 16.06.84.     702     read
cndx 24          1130 ut 16.06.84.     292     read
cndx 25          1149 ut 16.06.84.     125     read
hg5bg            1154 ut 16.06.84.     61      open
cndx             1535 ut 16.06.84.     569     read
hg5c1            1737 ut 18.06.84      194     open
e hg5c1          2220 ut 18.06.84      71      read
cndx 26          0519 ut 18.06.84      540     read
hg4zc            0921 ut 18.06.84      42      open
news 44          1425 ut 19,06.84      183     read
total of 38 messages using 18477 bytes.  14291 bytes remain.
1500 ut 19,06.84  next?
```

3. ábra. Tárolt információk teljes tartalomjegyzéke

betartva, azokat végrehajtja, illetve válaszol rájuk.

Fő feladata központilag szerkesztett friss információk és felhasználói üzenetek tárolása. Tárkapacitása 32 kbájt. Utasításai egyszerűek, könnyen elsajátíthatók. Ezeket akkor hajtja végre, ha aktív állapotban van. Ez egy megfelelő, mindenki által ismert kódszó megadásával történik, amely esetünkben a .rita szó. Ez az eljárás látható az *1. ábrán.*

Az első sor helyezte üzembe a számítógépet, a többi sor pedig annak válasza. Hogy éppen milyen információk találhatók a gépben, azt egy tartalomjegyzékből tudhatjuk meg, mely az egyes fájlok nevét tartalmazza. A könnyebb eligazítást segíti, hogy megállapodás szerint ezek a nevek utalnak a tartalomra. Az egyes hírcsoportok tartalomjegyzéke külön-külön fájlban található, melynek neve állandó. Így elegendő csak azt lekérni, amire kíváncsiak vagyunk, és nem kell kivárni sok, számunkra érdektelen anyagot.

A legjellegzetesebb hírblokkok a terjedési információk, sajtóreferenciák, rövidebb BASIC programok listái, hirdetések stb. Egy ilyen jellegzetes információ lekérdezését mutatja a *2. ábra.*

Mi kell a rendszer használatához? A rádión kívül, amely alkalmas a 145 MHz-es amatőrsávban, FM üzemmódban történő munkára, elsősorban egy terminálra van szükség, amely fogadni tudja az ötbites távgépírókódot.

Erre a célra legalkalmasabb egy személyi számítógép, amely minimális hardverkiegészítéssel és megfelelő programmal jól használható terminállá alakítható. Az állomások zöme már ezt az utat választotta, és egyre kevesebben forgalmaznak mechanikus távgépíróval. A hazai gépparkot figyelembe véve, a BME Rádióklub kollektívája elsőként ZX81-re és ZX-Spectrumra dolgozta ki a szükséges elemeket, de elkészült a HT-1080Z iskolaszámítógépre is.

A program a fejlesztési idő lerövidítése és a más gépeken való könnyű implementálás érdekében az iskolaszámítógépre kifejlesztett zFORTH nyelven íródott. Használata rendkívül gyors működése miatt számos olyan programozási fogás alkalmazását tette lehetővé, amelyre BASIC-ben nem nyílik mód. A vétel és az adásra előkészített szöveg szerkesztése egymástól függetlenül, egyidőben történik, osztott képernyőn. A szövegszerkesztést screen-editor támogatja. A leggyakrabban használt üzenetek előre programozhatók és tetszőleges mélységben egymásba ágyazhatók. A program tárolja az utolsónak vett 32 sort, amelyek a vétel zavarása nélkül visszalapozhatók, kazettára kimenthetők.

A programon kívül szükség van egy soros interfészre és egy egyszerű modemre, amely a digitális jeleket olyan alakúra hozza, hogy rákapcsolhatók legyenek az adóvevőre, illetve visszaalakítja az onnan jövőket.

Reméljük, ezzel a RADIR-ba újabb felhasználók kapcsolódhatnak be. Elsősorban azokra az iskolákra, művelődési házakra gondolunk, ahol már működik rádióklub, és a mikrogépek is rendelkezésre állnak.

Természetesen a rendszerbe való bekapcsolódás legfontosabb alapfeltétele a rádióamatőr adóengedély, mert csak ennek birtokában használható a már említett 145 MHz-es amatősáv. Az engedély megszerzéséhez a területi rádióklubok tudnak segítséget nyújtani.

A RADIR-ral összefüggésben álló bármilyen ötlettel, kéréssel a BME Rádióklubhoz lehet fordulni. Címe: Budapest, Goldman György tér 3. 1111.

A jövő

A leírtak az első kezdeti lépéseket jelentik egy hosszú úton. Az ehhez képest minőségileg újat adó adatátviteli hálózatok első nemzetközi szabványai már megfogalmazódtak, és átültetésük a gyakorlatba hazánkban is folyik.

Várhatóan néhány éven belül pályára állnak azok a rádióamatőr-műholdak is, amelyek összekapcsolják a nagy távolságban levő, ma még elszigetelt kisebb rendszereket nagyobb, kontinenseket összekötő hálózattá. Ez ma már nem utópia. S hogy mikor valósul meg, az rajtunk is múlik.

MÁRKUS BÉLA
Budapesti Műszaki Egyetem MHSZ-KISZ Rádióklub

nak. Ismerkedjünk meg ezzel közelebbről.

1983 eleje óta működik Galyatetőn az első hazai digitális átjátszó, amely két, egymástól független csatorna átvitelét biztosítja 50 Baud sebességgel. Állandó jelleggel tudják használni például budapesti, veszprémi, debreceni és egyes csehszlovák amatőrök is. Eddig mintegy száz különböző állomás forgalmazott segítségével.

Ezek közül az egyik fontos állomás a RITA, a Rádióamatőr Információkat Továbbító Adatállomás, amely valójában egy erre a célra kifejlesztett mikroszámítógép. A csatornán folyó forgalomban felismeri a neki szóló utasításokat, és a rádióamatőr-forgalom szabályait

```
hg5bme de hg0dy
 .read ucsw

ucsw                1629 ut 31.03.84.    197    read

   a mikroszamitogepes  infok  tartalomjegyzeke

 ucsw 1  -  a mikrogepes infok celja, tartalma
 ucsw 2  -  tavolsagszamitas 1k-as zx81-en

  'x'  es  'y'  jelenteset lasd az  intr 12  blokkban

1501 ut 19,06.84  next?
```

4. ábra.

```
hg5bme de hg0dy
 .exit ----- bye

mso deactivated 1508 ut 19,06.84
 bye
```

5. ábra. Kilépés

Szki

Olcsó, azonnal vásárolható,
szerződéskötéssel átvehető,
kompatíbilis szoftverrendszerek

Szövegkezelés és feldolgozás
Műszaki tervezés, méretezés
Matematika, statisztika
Adatrendezés
Fájlszervezés
Ipari és mezőgazdasági célokra

Sci-L

Fővállalkozás
Kereskedelem
Vevőszolgálat

1011 Budapest, I., Iskola u. 10.
Levélcím: 1251 Budapest, Pf. 19.
Telex: 1.22–5381
Telefon: 260-000

Scitel

Fővállalkozás
Bérletezés
Telematika

1015 Budapest, I., Donáti u. 35–45.
Levélcím: 1251 Budapest, Pf. 19.
Telefon: 650-122

LÉPÉSRŐL LÉPÉSRE

A számítógépes sakkprogramok játékereje szinte napról napra, a szemünk láttára nő. A kezdeti, nevetségesen gyengén játszó programok után napjainkra a legjobb sakkprogramok a játszma legtöbb fázisában elérték a magyar mesterjelölti minősítésnek megfelelő szintet. A mikroszámítógépek budapesti sakkvilágbajnoksága és a New York-i számítógépes világbajnokság játszmái is a sakkprogramok folyamatos javulásáról tettek tanúbizonyságot.

A látványos eredmények a számítógép-tudomány rohamos fejlődését sejtetik. Mint látni fogjuk, nem egészen ez a helyzet. A sakkprogramok játékerejének fokozódása nagyrészt a technika fejlődésének (a mind gyorsabb, nagyobb és olcsóbb számítógépek megjelenésének) köszönhető; a jelenlegi legerősebb sakkprogramok is nagyon hasonló elvekkel dolgoznak, mint a legelső, kezdetleges programok. Az elvek megvalósításának technikája lényegesen fejlődött, de új elvek nemigen hódítottak tért.

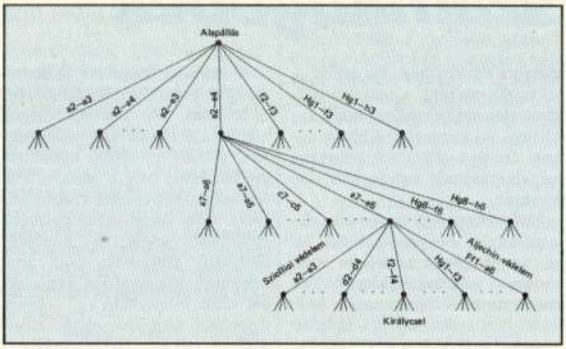
Sok finomítás történt és várható a jövőben is. De mert az alap változatlan, annak, aki most kezd el foglalkozni a sakkprogramozással, s részt szeretne venni a lapban kiírt pályázaton, nem kell attól tartania, hogy a fejlődés pontos ismerete nélkül nagy munkával rossz irányban indul el.

Cikksorozatunkban – hogy a pályázni kívánókat is segítsük – bemutatjuk azokat az eljárásokat, amelyekkel napjaink sakkprogramjai dolgoznak. Elemezni fogjuk azokat a lehetőségeket, amelyek ezen algoritmusok keretein belül további fejlődés lehetőségét vetítik előre, és vizsgálni fogjuk ezeknek az eljárásoknak a korlátait is. Látni fogjuk, hogy egy sakkmester alapvetően másképpen gondolkodik, és ezzel máris a mesterséges intelligencia legáltalánosabb problémáihoz jutunk el: hogyan lehetne ezt az egészen másfajta gondolkodásmódot számítógépen megvalósítani. Előbb azonban ismerkedjünk meg az alapokkal.

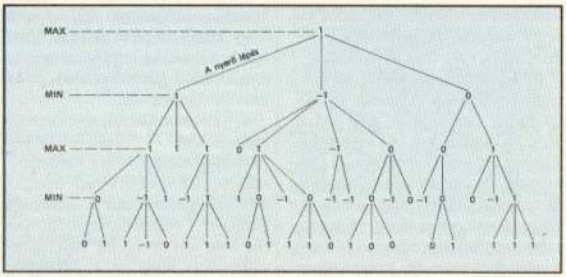
A játék fája

Egy adott kiindulási állásból lejátszható összes lehetséges játszmafolytatás grafikusan ábrázolható. Jelöljük a kiinduló állást egy ponttal.

Tegyük fel, hogy ebben az állásban világos van lépésen. Az összes lehetséges szabályos világos lépés után létrejövő állásokat jelöljük egy-egy újabb ponttal, és a kiinduló állást jelölő pontot kössük össze mindezekkel a pontokkal.



1. ábra. A sakkjáték fájának kezdete



2. ábra. A minimax algoritmus alkalmazása egy egyszerű fára

Ezt utóbbi pontok mind olyan állást képviselnek, ahol sötét van lépésen. Minden ilyen álláshoz az ott lehetséges összes szabályos sötét lépés után létrejövő állásokat megint jelöljük egy-egy ponttal. Kössük össze ezekkel a pontokkal mindig azt a pontot, amely azt az állást ábrázolja, amiből az illető állás következett.

Ismételgessük ezt az eljárást mindaddig, amíg olyan állás nem jön létre minden ágon, amelyikből már több lehetséges lépés nem ágazik ki. Ezek a pontok felelnek meg például a matt- és patt-állásoknak.

Ha a kiindulási állás az alapállás volt, akkor ezzel a módszerrel az összes lehetséges sakkjátszmát feltérképeztük *(1. ábra)*. Az így létrehozott diagramot nevezzük a *játék fájának*. A sakkjáték fája biztosan véges, mert minden állásban csak véges számú sok szabályos lépés képzelhető el, és a játszma hosszának is eleve korlátai vannak a szabályok miatt. (Ötven lépés ütés vagy gyaloglépés nélkül eleve döntetlent eredményez, és az ütések és gyaloglépések száma korlátozott.) Most megadunk egy *elméleti* algoritmust, amellyel ha egyszer birtokunkban van a játék fája, meghatározható, hogy hogyan játsszon egy abszolút jó játékos.

A minimax algoritmus

Mit is várhatunk el egy ilyen abszolút jó játékostól? Mi történik, ha egy másik abszolút jó játékossal kerül szembe? Egy abszolút jó játékost úgy képzelhetünk el, hogy minden állásból kihozza a maximumot. Tehát ha egy állás nyerhető, akkor nyer, és csak akkor veszít, ha mindenképpen veszítenie kell.

Az, hogy egy állás *nyerhető*, azt jelenti, hogy az ellenfél akármit is húz, válaszul ismét olyat lehet lépni, hogy az állás továbbra is nyerhető maradjon. Azaz, hogy az előző mondat rendületlenül ismételgethető mindaddig, amíg matthoz nem jutunk.

Hasonlóan gondolkodhatunk akkor, ha arról van szó, mit jelent az, hogy egy állásból *döntetlen* hozható ki. Az pedig, hogy egy állásban *veszíteni* kell, azt jelenti, hogy akármit is lépünk, olyan állás jön létre, amely az ellenfél számára nyerhető (az előbbi értelemben).

Azt mondhatjuk, hogy az elképzelt abszolút jó játékos az, aki a fenti gondolatmenetet a sakkjáték fájának teljes sűrűjén végig tudja vezetni. Ez a végigvezetés nemcsak elméletileg lehetséges, de még csak nem is nagyon nehéz feladat; megoldására bemutatunk egy egyszerű eljárást.

Rendeljünk a játék fájának minden pontjához egy-egy számot a következő szabály szerint: a végállásokat ábrázoló pontokhoz 1-et, ha ott világos nyert, 0-t, ha az eredmény döntetlen és −1-et, ha ott sötét nyert. Ezután haladjunk fölfelé a fán úgy, hogy sorra értéket adunk az olyan pontoknak, amelyekre az összes olyan pontoknak, amelyekkel az éppen vizsgált pont lefelé össze van kötve, már adtunk értéket.

Az értékadás szabálya a következő: ha a kiértékelendő pontban *világos* van lépésen, akkor az összes, vele összekötött pontokhoz rendelt értékek *maximumát* rendeljük a szóban forgó ponthoz, ha pedig ott *sötét* van lépésen, akkor az összes, vele összekötött pontok értékeinek *minimumát* adjuk értékként *(2. ábra)*.

Ezt az eljárást folytatjuk mindaddig, amíg értéket adunk a kiinduló állást ábrázoló pontnak is. Könnyen látható, de azért érdemes egyszer végiggondolni, hogy a kiindulási pont akkor kap 1-es értéket, ha a kiindulási állás világosnak nyerhető (az abszolút jó játékos számára), 0-ás értéket akkor kap, ha a kiindulási állásból döntetlen érhető el, és −1-et akkor rendelünk a kiindulási álláshoz, ha azt egy abszolút jó játékos ellen mindenképpen el kell veszíteni.

Az eddigiekben leírt eljárást nevezzük *minimax* algoritmusnak. Ennek ismeretében a tökéletesen sakkozó számítógép létrehozása nagyon egyszerű: a gépnek csak fel kell állítania minden adott állásban a játék fáját, végre kell rajta hajtania a minimax algoritmust, és egy olyan lépést választania, amelynek megfelelő ponthoz a fában 1-es érték van rendelve. Ha ilyen lépés nincs, akkor döntetlenre vezető lépést kell választania.

Ha ez ilyen egyszerű, akkor miért nem így csinálják?

MÉRŐ LÁSZLÓ

(Folytatjuk)

A μ M megkérdezte Szabó László nemzetközi nagymestert, hogyan vélekedik a számítógépes sakkról. Az alábbiakban őt idézzük.

Sakkozó számítógépek

Zavarban vagyok. Mert hogy jövök én, a sakkozó, ahhoz, hogy egy számítástechnikai lapban véleményt mondjak a sakkozó számítógépek lehetőségeiről? Bár tizenhat éves fiam révén van valami fogalmam e gépek működéséről, s magam is próbálkoztam már egy-egy játszmát váltani velük, de az ezzel kapcsolatos kevés tapasztalat aligha jogosíthat fel arra, hogy érdemben hozzászóljak a kérdéshez. Ám egy könnyelmű pillanatomban igent mondtam a lap szerkesztőjének ...

A sakkvilág két kiválósága, dr. Max Euwe és Mihail Botvinnik egymással homlokegyenest ellenkező véleményen volt egy tökéletesen sakkozó gép megalkotásának lehetőségéről. Euwe, a matematika egyetemi tanára (!) nem hitt ebben. Botvinnik viszont, miután szakított a gyakorlati sakkozással, a leghatározottabban állította, hogy ilyen masinát lehet készíteni. Hozzá is kezdett egy program kidolgozásához, de az eredmény egyelőre nem az ő igazát bizonyítja.

Ez nem jelent semmit – mondják egyes szakértők. Véleményük szerint ugyanis Botvinnik egyrészt rossz úton indult el, másrészt segítsége sem volt olyan mértékű, amilyenre szüksége lett volna. Az ő sikertelenségének már csak azért sincs bizonyító ereje – mondják –, mert az utóbbi évtizedben mások is kedvet kaptak sakkprogramok kidolgozásához, és mint azt a mikroszámítógépek budapesti és a nagyobb gépek New York-i világbajnoksága bizonyította, csodálatos eredményeket értek el.

Ennek ellenére szerintem az elért színvonal még igen messze van attól, amit az ember a fejében levő „mikroszámítógéppel" akár jelenleg is produkálni tud. Mi, nagymesterek sem tudjuk elmondani, hogyan képes az agyunk jóformán azonnal, az állás megpillantásakor kiszűrni azokat a lehetőségeket, amelyek nem jöhetnek számításba, és csupán a fennmaradó egyet vagy néhányat mérlegelni.

A gép persze sokkal gyorsabb lehet a változatok számításában, mint az emberi agy. De vajon a sakkozásnak csak a matematikai része elegendő-e ahhoz, hogy kiválóan sakkozzunk? Ebben érzem én egy tökéletes program megalkotásának legnagyobb nehézségét.

Elsősorban arra szeretnék választ kapni, hogyan építhető be egy programba a küzdelem pszichológiai oldala? Eddig kérdésemre senkitől sem kaptam kielégítő választ, holott számos játszma nem a legjobb, sokszor matematikai pontossággal kiszámítható lépésekkel, hanem kerülő úton, az ellenfél pszichéjében rejlő negatív tulajdonságok kihasználásával vezet győzelemre.

Ez természetesen abból adódik, hogy tökéletesen számoló, sakkozó ember nem létezik, még akkor sem, ha minden kor esetleg a tökéletesség illúziójában él. Ma jól sakkozunk, de száz év múlva bizonyára mosolyt fakaszt majd a mai játéktudásunk, mint ahogy mi is mosolygunk, amikor a híres 1843. évi párizsi Staunton–St. Amant páros mérkőzés tornakönyvében azt olvassuk, hogy „... mindkét úr nemcsak a jelenleg élő legerősebb sakkozókhoz tartozik, hanem a *sakkozás elméletének legtökéletesebb ismeretével* rendelkeznek". Ez az egyik szempont, amely bennem kételyeket támaszt.

A másikat magában a matematikai programozási lehetőségben látom. „... a tétlen világos huszár cserélődik egy védő tisztért, a csere minőségi különbséget hoz létre. Míg a csere, a tisztek megtanult értékét (futó = huszár) véve figyelembe, egyszerű egyenleget jelent, a minőségi (helyzeti) különbség csak igen bonyolult matematikai képletekkel volna kifejezhető ..." – írtam legutóbbi könyvem egyik elemzésében (50 év 100 000 lépés). – De kifejezhető-e egyáltalán?

Talán akad valaki, aki e cikkem nyomán indíttatva érzi magát, hogy kételyeimet eloszlassa. Vagy majd csak akkor kapok választ, ha megjelenik a piacon az első „tökéletes" sakkprogram?

Dr. Lindner László, a rovat vezetője válaszol:

A nagymester olyan véleménynek és kételyeknek ad hangot, amelyek általánosak a sakkozók körében. Felvetésük tehát időszerű, még akkor is, ha egyértelmű válaszokat adni és a kételyeket eloszlatni (ma még?) nem lehet.

A sakkal hivatásszerűen foglalkozók és a sakkot értők nézeteivel s azok változásával azonban kell foglalkozni. Mivel ők sakkoznak a legjobban, az általuk támasztott követelményeket a programozóknak nemcsak figyelembe kell venniük, de iránymutatónak kell tekinteniük, ha az a céljuk, hogy a számítógépek elérjék, sőt túlszárnyalják az emberi sakktudást.

Érdekes jelenség, hogy kevés sakkozó hisz ebben, s az is, hogy kevesen foglalkoznak közülük sakkprogramozással. Botvinnik professzor a „nagy" kivétel, bár rajta kívül is akad néhány kiváló mester a világ kiemelkedő sakkprogramozói sorában. Mindenekelőtt David Levy professzor, egykori levelezési világbajnok. David Levy – ez közismert történet – 1968-ban nagy összegű fogadást kötött, hogy őt tíz éven belül számítógép nem győzi le, sőt, ezt további öt évre meg is hosszabbította. Amikor 1982-ben Budapesten megkérdeztük őt, mi most a véleménye a sakkprogramozás jövőjéről, azt felelte: semmi kétsége afelől, hogy a számítógép tudása előbb-utóbb túl fogja szárnyalni a világbajnokét.

Persze ez a vélemény nem csalhatatlan, de arra jellemző, hogyan változott a kérdésről az egyik legjobb szakértő nézete. Egyébként változott Euwe professzoré is; alig néhány hónappal halála előtt beszélgettem vele utoljára, és a számítógépes sakk lehetőségeit „beláthatatlannak" mondta.

De elsősorban Botvinnik fontos szerepét akarom pontosítani. Nem

a gyakorlati sakkozással való szakítása után állította, hogy el lehet készíteni egy sakkozni tudó masinát. Hiszen 1963-ban, amikor Petroszjan elhódította tőle a sakkvilágbajnoki címet, s ő dolgozni kezdett Pionir sakkprogramján, már jócskán léteztek sakkozni tudó számítógépek. Botvinnik továbbra is játszott versenyeken; 1969-ben 1–2. volt Wijk aan Zee-ben.

Igaz, hogy a sakkprogramozással nehezen haladt, mert mérnöki munkája során más számítástechnikai feladatokat kapott. De tévedés, hogy rossz úton indult volna el! Botvinnik alapvetően a „B" stratégia hive, és éppen a mikrogépek térhódítása mutatja, hogy korántsem mondták még ki az utolsó szót az „A" és „B" stratégia hívei közötti vitában. Botvinnik a számítógépes sakknak az egész világon elismert szakértője. Első munkáját 1970-ben publikálták angol nyelven, és ezt újabbak követték. Példája azt a gondolatot ébreszti, hogy a legjobb programot talán valamikor éppen egy világmester készíti majd el.

Egyetértek a nagymesterrel abban, hogy a számítógép sohasem fog „pszichológiai sakkot" játszani. De nem arról van szó, hogy a sakkozás matematikai részét képes az emberi agynál tökéletesebben fejleszteni. A tétel így hangzik: a programozó a sakkbeli elemeket matematikai egységekre „számítja át", s ennek alapján értékeli az állásokat és dönt a soron lévő lépés felett. Nem kell bonyolult matematikai képlet ahhoz, hogy a számítógép ne csak a bábok anyagi erejét értékelje, hanem egyéb tényezőket is: mi azok hatóereje, milyen mezőket tartanak megszállva, milyen vonalak-átlók nyílnak meg; sőt: hogyan alakul a gyalogok formációja vagy a király biztonsága. A probléma megoldása a sakkbeli kritériumok minél teljesebb számbavételén, helyes felismerésén és a számítás helyes sorrendjén múlik, hiszen a gépek óriási, nem tudni, meddig fokozható sebessége mellett is véges idő áll rendelkezésünkre.

A sakkprogramozást cikksorozatban ismertetjük olvasóinkkal. Örülnénk, ha minél több sakkozó – mester, nagymester – is akadna közöttük, hogy teljesebb képet nyerjenek a sakkprogramozás lényegéről, lehetőségeiről, perspektívájáról.

Dr. Úry László: Commodore 64 I., II. kötet (Bp. 1984. LSI ATSZ. 500 oldal. Ára 370,– Ft.)

Az OMIKK LSI Alkalmazástechnikai Tanácsadó Szolgálat kiadásában jelent meg Dr. Úry László „Commodore 64 – Basic felhasználói kézikönyve" c. kétkötetes műve. A Magyarországon is egyre népszerűbb mikroszámítógépnek kiterjedt irodalma van. Német és angol nyelvű kézikönyvek járnak kézről-kézre – jobbára másolásban –, s boldog az a Commodore-tulajdonos, aki így hozzájuthat ezekhez az egyébként igen drága könyvekhez.

Úgy látszik azonban, a gép egyértelmű világsikerével nem tud lépést tartani a vele foglalkozó irodalom. A kézikönyvek többsége tartalmilag és didaktikailag erősen kifogásolható, a bennük levő példaprogramokban gyakoriak a hibák. S mintha minden könyv azzal a szándékkal íródna, hogy (fáradságos) olvasásuk után, a csalódott felhasználó újabb és újabb könyveket legyen kénytelen megvásárolni. Ezért üdvözlésre méltó minden olyan vállalkozás, amelynek célja, hogy egyszerűen feldolgozható formában gyűjtse össze azokat az ismereteket, amelyekre az – akár kezdő, akár haladó – Commodore-felhasználónak munkája során szüksége lehet.

A szerző nem kevésre vállalkozott. Saját szavai szerint programozási kézikönyvet és programozási tankönyvet kívánt írni, amelyben a felhasználó felvetődő kérdéseinek többségére választ találhat. Az IBM-irodalmat ismerő olvasóban az a kép alakul ki, hogy ez a könyv nemcsak a programozói segédletet („Programmer's Guide") és a tankönyvet akarja egyesíteni, hanem még a hivatkozási kézikönyvet („Reference Manual") és a programlogikát magyarázó kézikönyvet („Program Logic") is. E feladattal szinte megvalósíthatatlan követelményeket állít a szerző saját maga elé, főként ha figyelembe vesszük, hogy egy ilyen könyv elsősorban nem szakemberek, hanem a számítástechnikával most ismerkedő vagy azt kedvtelésből űző felhasználók kezébe kerül. Ilyenkor a didaktikailag kristálytiszta felépítés, a szigorú logikai következetesség, a tartalmi és a formai kifogástalanság, a világos és egyértelmű megfogalmazás elengedhetetlen követelmény.

Sajnos, a szerző szándéka nem sikerült maradéktalanul. Ez nemcsak a maximalista célkitűzésnek a következménye, hanem annak is, hogy a könyv sietségnek és átgondolatlanságnak az érzetét kelti. Mintha a minél gyorsabb elkészülés és piacradobás kényszere hajtotta volna a szerzőt és a kiadót egyaránt, és így inkább csak a források követésére és összegyűjtására futotta az erőből, az előbb említett követelmények teljesítésére már nem.

A könyv tematikájában felölelt szinte a teljes idevágó területet. A Basic interpreter, s az egyes Basic utasítások végrehajtása (néhány Basic kiterjesztésre vonatkozóan is), a perifériális egységek működése és kezelése, a gépi kódú programozás, a grafikus és hanggenerálási lehetőségek, programozási tanácsok mind külön fejezetet kaptak. Talán hasznos lett volna, ha a gépről egy általános rendszertechnikai összefoglaló is készült volna, s az egyes fejezeteknek a hardverre vonatkozó fogalmai erre utalnának vissza. Így elkerülhető lett volna az a gyakran zavaró, a megértést megnehezítő helyzet, hogy itt is, ott is felbukkan a szövegben egy-egy fogalom, mintha ismerete a legtermészetesebb lenne, holott csak néhány lappal később vagy egy másik fejezetben (esetleg sehol sem) kerül megmagyarázásra.

Az olvasás nehézségéhez az is hozzájárul, hogy a könyvhöz nem készült jelmagyarázat. Persze az olvasó előbb-utóbb rájön, hogy például mit jelent a $300 kifejezés és miben különbözik a ($300) kifejezéstől, a gyakorlati szempontok azonban joggal igénylik a használt jelek, rövidítések rendezett magyarázatát. Ugyanide tartozik, hogy a szerző ugyanazt az angol szakkifejezést hol lefordította magyarra, hol nem. Így az olvasóra hárul, hogy a magyar és angol kifejezéseket összepárosítsa. A „Clear to send" és a „Nyugtázás az adónak" elnevezés például ugyanazt a vonalat jelenti az RS–232 csatornán.

A szerző a Bevezetésben azt ajánlja, hogy a kezdők először a 2., a 3., és a 10. fejezetet olvassák. Nem tudjuk, hogy kiket ért ő kezdőkön, de még a haladó C–64 programozási tanfolyamot „kezdőket" is inkább lebeszélnénk a 3.1. és a 3.2. fejezetekkel való foglalkozásról. (Miért lenne egy kezdőnek például arra szüksége, hogy programjainak memóriaképét hexadecimálisan a képernyőre kiírassa?)

Sajnos a könyv olvashatóságát tovább rontják a szövegben hemzsegő gépelési hibák. Ezért a recenzens sokszor nem tudja pontosan megállapítani, mi gépelési, mi pedig tartalmi tévedés. Érthetetlen például, hogy az egyik oldalon megadott képlet hogyan változhat meg azalatt, amíg az olvasó a következő oldalig ér? Vagy: legalábbis magyarázatra szorul, hogy hogyan lehet 380 és a 75 szám 16 bites megfelelője −241, ill. 15359.

A 8. fejezetben ígért függelékrész, a 8 oktáv hangjainak megfelelő regiszterértékekről, nem került be a könyvbe. A három Fourier-sorfejtési képlet megadása matematikailag nem egzakt, sőt kettő közülük hibás is.

A könyvben bőségesen vannak példaprogramok és programrészletek. A programok között azonban találunk olyanokat is, amelyek hibásan, vagy legalábbis nem a leírásnak megfelelően működnek.

Ennyi aggály és kifogás láttán felvetődhet a kérdés, hogy használható-e a gyakorlatban a könyv? A válasz csak azért igenlő, mert jelenleg ez az egyetlen magyar nyelvű, átfogó ismereteket tartalmazó kézikönyv a C–64-ről. Az olvasónak viszont fel kell készülnie arra, hogy alaposan meg kell küzdenie az anyaggal, s ehhez a küzdelemhez sajnos nem sok segítséget kap.

VÁRGEDŐ TAMÁS

Steven Vickers: BASIC programozási kézikönyvek. ZX81 és ZX–Spectrum. (Budapest, 1984. Ipari Informatikai Központ, 136 és 280 oldal)

Egymás között eddig gépkönyvnek hívtuk, mert a géppel együtt adták – igaz, angol nyelven. Most már hozzáférhetünk magyarul is: a vékonyabb könyv, a ZX81-eshez 320 forintért, a vastagabb, a Spectrumhoz 280 forintért kapható. A furcsa eltérés oka talán az, hogy az előbbi kiírón, az utóbbi írógépen készült; egyébként mindkettő egyszerű rotaprint eljárású sokszorosítás. Az árak nem befolyásolják a kiadványok kapósságát: ma például csak előjegyzést vesznek fel az Akadémia Könyvesboltban a Spectrum kézikönyvre.

A kiadványok alkalmazási segédletnek készültek „A mikroszámítógépek és alkalmazási rendszereik kutatása, fejlesztése" című OMFB tárcaprogram egyik alprogramjának keretében. Vegyük így: angol tudásunk hiányosságait pótló alkalmazási segédletek.

Előljáróban megjegyzem, hogy elismerést érdemel a fordító, aki közérthető, anglicizmusoktól viszonylag mentes fordítást készített, csakhogy kilétét nem lehet tudni.

A két eredetit összevetve a Spectrum kézikönyv (1982-ben adták ki) személytelenebb, fordítása talán így könnyebb volt; a ZX81 (1980-as az első kiadás) személyes hangvételét nem mindig sikerült a fordítónak átmentenie, bár nálunk közismerten egy sokkal szárazabb szakkönyvstílus az elfogadott, mint az angolszász irodalomban, így gyakran nem könnyű fordítói megoldást találni.

Két szempontot veszek tehát figyelembe: magukat az eredeti műveket és a fordítást.

Úgy hiszem, sokan vannak, akik nap mint nap forgatják akár a magyar, akár az eredeti könyvet, és többnyire nem hiába, mert minden lényeges BASIC programozási kérdésükre választ is kapnak. Sőt jó néhányan lehetnek azok is, akik éppen ezekből a könyvekből tanulják meg a BASIC-et, „a kezdők általános célú szimbolikus utasításkódját". Mindkét célra alkalmasak a könyvek. Természetesen nem önmagukban, hanem a géppel együtt, amint ezt a ZX81 bevezetője így hangsúlyozza: „A számítógépet mindig használjuk! Ha olyan kérdésünk van, hogy mit fog csinálni, ha ezt vagy azt tesszük, a válasz így nagyon egyszerű lesz: beírjuk és megnézzük. Ha a kézikönyv arra utasít, hogy írjunk be valamit, mindig gondoljuk meg, hogyan oldhatnánk meg másképpen a feladatot, és próbáljuk ki saját megoldásunkat is. Minél több saját programot írunk, annál jobban megértjük a ZX81 működését. (Nem ezt a módszert nevezik programozott tanulásnak.)" – Hozzátehetjük: ez a felfedező tanulás módszere.

Tehát gép nélkül még jó könyvből sem, még magyarul sem lehet valóban elsajátítani a programozást.

Mennyiben tér el az eredeti és a fordítás? Példaként a Spectrum kézikönyvet vegyük szemügyre. Az egyetlen látványosabb különbség, hogy a D függelékben a mintaprogram fordítása helyett a programok magyarázata található, az eredeti könyvre vonatkozó utalásokkal.

Akad hasznos kiegészítés is, mint például a # képernyőn alkalmazásáról (a 15. oldal 2. gyakorlatában), ami valóban hiányossága az eredetinek. Érdemes lett volna mást is javítani: némileg félrevezető a DATA, READ, RESTORE használatát bemutató program (6–3. oldal a fordításban, 42. oldal az eredetiben), mert bár a magyarázó szöveg pontosan eligazít, a RESTORE 10 helyett didaktikusabb RESTORE 60 jobb lett volna. Hasonlóan a 12–2. oldal példaprogramjában is van egy kis következetlenség, nevezetesen hiányzik az 5 DIM(10), hiszen az előző szöveg nem utasít egyértelműen a direkt utasítás beírására.

Más jellegű apróságok: olykor le van fordítva a program szövege, olykor nem, néha pedig hiányzik belőle az a látens humor, amely akaratlanul is megmosolyogtatja az embert. Mert nem ugyanazt váltja ki az a kérdés, hogy „Olvastad-e az Egri csillagokat?" (hiszen ki nem?), mint az, hogy „Befejezted-e már a Finnegans Wake-et?" (mert ugyan ki?). (Joyce regénye egyetlen mondatban örökít meg egy álom és ébrenlét közötti állapotot.)

Szórakoztató volna összegyűjteni azt a számítógépes–angol–magyar zsargont, ami iskolákban, klubokban és más programozó társaságokban kialakult. Néhány általam ismert példa: „kiNEWzom a programot", vagy RUNi neki!". Mégis jobb lett volna „ink szín" és „paper szín" helyett tinta- és papírszínt fordítani a kézikönyvben, az umlautos o helyett pedig egyszerűen „ö"-t írni (16–17. oldal).

A Sinclair gépek a számítógépek között új irányt indítottak el: könnyen kezelhetőségükkel és olcsóságukkal kaput nyitottak a felhasználók és ismerkedni vágyók széles tömege előtt. Így nem meglepő, hogy legelterjedtebbek hazánkban is, bár nem hivatalos kereskedelmi utakon kerültek be az országba. A kézikönyvek is a nyitottságot, közérthetőséget támogatják. Ezért jó a magyar változat is.

Jó könyvek, érthető fordításban, csak néhány gépelési hibával, csak egy-két elcserélt és duplikált oldallal, bár kissé borsos áron. Talán azt is remélhetjük, hogy a kiadványok hatásaként a magyar személyi számítógépek kézikönyvei is ilyen jó felépítésűek, érthetőek, következetesek és kellemesek lesznek.

VOTISKY ZSUZSA

„Érted haragszom, nem ellened"

Olvasóink előtt minden bizonnyal nyilvánvaló, hogy lapunk elkötelezte magát az iskolaszámítógép-program mellett, és szorosan együttműködik a program gazdájával, a Tudományszervezési és Informatikai Intézettel. Teszi ezt azért, mert a szerkesztőségnek és munkatársainak meggyőződése, hogy a számítógépekkel kapcsolatos ismereteknek az általános műveltség szerves részévé kell válniuk, legalább annyira, mint az anyanyelvi ismereteknek vagy a kétszerkettőnek.

A program végrehajtása kezdettől fogva erőltetett menetben folyik, s akik lelkesen és odaadóan dolgoznak sikeréért, azokat még kergetjük, olykor bántjuk is. Jó magyar szokás szerint kritizáljuk a kiválasztott gépet, sokalljuk az árat, kéveselljük a rendelkezésre álló oktatóprogramokat, elégedetlenek vagyunk a megbízhatósággal stb. stb. Nos, bírálni csak azt lehet igazán, ami már van, él vagy működik, s az iskolaszámítógép is ilyen, ráadásul az ország valamennyi középiskolájában. Hibái, hiányosságai természetesen vannak és lesznek is, sokuknak a gyártók és fejlesztők is tudatában vannak, s mindent megtesznek kiküszöbölésükért.

A hibákat és erényeket sok olvasónk a gép használata közben nyilván maga is észrevette, mások az Ötlet/Bitlet hasábjain találkozhattak velük. Rovatszerkesztőnk, dr. Simonyi Endre is olvasta az Ötlet 1983. novemberi számában megjelent, HT-1080Z-nek szentelt Vallatót, ugyanakkor maga is kínpadra vonta a gépet, lévén – sok más egyéb között – tapasztalt gépépítő, a SIMON 68 elnevezésű mikroszámítógép konstruktőre. Hozzászólását a HT-1080Z vallatásához az Ötlet/Bitlet 1984. február 2-i száma közölte. S ahogy az már ilyenkor lenni szokott, vita alakult ki a lap hasábjain, hozzászólás hozzászólást követett a március 1-i és 29-i, valamint a május 31-i Bitlet-ekben.

Érdeklődéssel olvastuk az érzelmektől, indulatoktól nem mentes vitát, úgy érezve, hogy amiről – ügyről és gépről – ennyit és így vitatkoznak, az csak jó lehet. Szívesen követtük volna tovább is a pro és kontrákat, ám hiába kerestük ezeket az Ötlet újabb számaiban, május 31-vel, Novák Ferenc hozzászólásával a szerkesztőség – úgy tűnik – levette a kérdést a napirendről. Kár! Mármint, hogy a vita – összegzés nélkül – abbamaradt. Ráadásul van néhány olyan momentum, amely vita tárgyát nem képezheti, rögzítésük így ránk maradt. Vegyük sorra őket!

Dr. Simonyi Endre vagy SIMON 68 nevű gépe nem vett részt az Iskolaszámítógép-pályázaton.

A HT-1080Z – különösen a kazettás tárolójú – megbízhatóságát illetően megoszlanak a vélemények. A gép betűkészletéből a magyar ábécé ékezetes karakterei közül továbbra is hiányoznak az Í, Ó, Ő, Ú, Ű, kisbetű méretű az Á és az É. A szöveg még nagyképernyős tévén is (64 karakter/sor sűrűséggel) kijelezve csak közelről olvasható, míg 32, ill. 16 karakter/sor sűrűséggel kijelezve az oldal függőlepenyős irányban félbe/negyedbe vágva jelenik meg.

A SIMON 68 betűkészlete megegyezik a teljes magyar ábécéével és írásmódja (a betűk nagysága és elhelyezése) megfelel a helyesírási szabályoknak.

A TRS 80 Model III 16 k kiskereskedelmi ára 1981 őszén 835 amerikai dollár volt (lásd Byte magazin, 1981. dec. 457. oldal). A Video-Genie I kiskereskedelmi ára ugyancsak 1981 őszén már csak 575 amerikai dollár volt (lásd Byte, 1981. dec. 427. oldal). A TRS 80 Model III nem ekvivalens sem a Video-Genie I-gyel, sem a HT-1080Z-vel. A különbség – a kidolgozás minőségén túl –, hogy a TRS-gép dobozába ún. videomonitort, diszkillesztőt (tápegységgel, kábelezéssel együtt), nyomtatóillesztőt és buszvédelmet is beépítettek.

A Video-Genie I nagykereskedelmi ára (1000 db-os rendelés esetén) kb. 30–40 százalékkal kevesebb lett volna a kiskereskedelminél, vagyis diszkillesztővel (de tápegység és kábelezés nélkül) és nyomtatóillesztővel 345–405 dollár, bővítmények nélkül – vagyis a HT-1080Z-hez hasonló műszaki paraméterekkel – 205–280 dollár, s még mindig 1981 decemberében.

Az első HT-1080Z-kbe épített import alkatrészek értéke 250 amerikai dollárra rúgott. Ma már a devizaköltség 100 dollár alatt van, és jövőre tovább csökken.

A következő sorozatban gyártott iskolaszámítógépek operatív kapacitását 64 kbájtra bővítik, ára változatlan marad.

Ezek a tények, érvek nélkül. Sapienti sat, a bölcsnek (ennyi is) elég, s így olvasóinknak is.

KÖNYVES TÓTH PÁL

Családi számítógépek

A személyi számítógép és a házi számítógép után hamarosan új fogalommal kell megbarátkoznunk, a családi számítógép fogalmával. Ennek feladata, hogy a család valamennyi tagja igénybe vehesse, számukra információt, szórakozást és esetleg oktatási lehetőséget biztosítson. A megfogalmazott kritériumokból következik, hogy ezek viszonylag nagy teljesítményű gépek – legalább 64 kbájt operatív tár, mágneses háttértár, nyomtató tartozik hozzájuk, és újabban a telefonhálózathoz csatlakoztatható modem. Ez utóbbi lehetővé teszi, hogy a családi számítógép nagyobb számítógépes hálózatokhoz csatlakozzon, ami intelligenciáját rendkívül megnöveli.

Nő a lemezegységek forgalma

A nehéz gazdasági helyzet ellenére a merev lemezegységek forgalma 1981-ben és 82-ben is 24 százalékkal nőtt, és elérte a 8 milliárd dolláros világpiaci eladást, 1985-re pedig 15 milliárd dolláros forgalmat jósolnak. A cserélhető lemezes tárolók népszerűsége csökken; 1985-re a piacnak csak 15 százalékát teszik ki. A leggyorsabb növekedés az 5 1/4 collos 30 Mbájt alatti fix Winchestereknél várható, elsősorban a személyi számítógépek terjedésével: az 1985-re várható igény 1 millió darab. Az IBM 3380 típusú lemezegységéből 1982-ben 11 ezer darabot adtak el, 1983-ban pedig kb. 28 ezer darabot.

Új IBM piac a Távol-Keleten

Az IBM Japan annak lehetőségét tanulmányozza, hogy 1984-től személyi számítógépeket ad el Dél-Koreának, Tajvannak, Hongkongnak és Szingapúrnak. Az IBM Multi-station 5550 néven ismert személyi számítógépet az IBM Japan fejlesztette ki, és 1983 közepén jelent meg 3 millió jen alatti áron. A közeljövőben 50 ezer darab személyi számítógép eladását tervezik az említett piacokon.

Szövegfeldolgozás Xerox kisgépen

A Xerox cégnek a 8010 Professional Information System kisszámítógépe 1983 eleje óta alkalmas japán szövegfeldolgozásra. A cég fejlesztői jelenleg azon dolgoznak, hogy az orosz nyelvű szövegek feldolgozására alkalmas szoftverüket kialakítsák.

Kié a mikropiac?

Az USA-ban már több, mint 100 asztali számítógép-értékesítő üzlet van, amelyek mintegy 200 különböző típust hoznak forgalomba. Ez azt eredményezi, hogy az üzletemberek számára mind nehezebb ésszerű számítógép-beszerzést végezni. Így az USA-ban olyan vállalatok alakulnak, mint a Businessland, amely az üzletemberek igényeit kívánja kielégíteni. A Businessland 24 millió dolláros tőkét szerzett, és csak 1983 első felében 6, második felében újabb 10 üzletet nyitott meg. 1985-ig üzletei számát 150-re kívánja növelni; ebből Európában 40 fog működni.

A Businessland tapasztalatai szerint a piacon nem azok a vállalatok tudnak leginkább érvényesülni, amelyek gyorsabb, kisebb vagy jobb hardvert tudnak előállítani, hanem azok, amelyek hatékonyabb kereskedelmi hálózatot tudnak kiépíteni.

Csökkenő árak az USA-ban

1983-ban a személyi számítógépek piacának forgalma 8 milliárd dollárra becsülhető. Egyre több gyártó vállalat folytat küzdelmet azért, hogy minél nagyobb részt biztosítson magának ezekből a lehetőségekből. Ezzel párhuzamosan a személyi számítógépek ára csökken.

Az árcsökkentésben a legagresszívabb a Commodore. A VIC–20 berendezést a nagykereskedelemben már 90 dollárért adják, pedig a katalógus szerinti ára meghaladja a 150 dollárt.

Mikroszámítógépek a főutcán

A mikroszámítógépek kereskedelmének újabb változata van születőben: a főutcai számítástechnikai üzlet. Célja, hogy a kis- és középvállalatok számára támogatást nyújtson az olcsón forgalomba hozott számítógépek lehetőségeinek jobb kihasználására. Ez a főutcai üzletrendszer azonban bizonyos veszélyeket is rejt magában. A tanácsadás ugyanis nem függetleníthető teljesen egy adott berendezéstípustól, és így könnyen megtörténhet, hogy ezek az üzletek végső soron gép-ügynökökké válnak.

A 32 bitesek forradalmat hoznak?

Várható, hogy a 32 bites gépek piaca gyorsabban fog fellendülni, mint a 16 biteseké. Ezek asztali változatai már valóságos forradalmat jelenthetnek, mivel teljesítményük a szupermini gépekhez hasonlítható, áruk viszont azok árának csupán 1–5 százaléka. Ezek a gépek várhatóan 1985-ben törnek be a piacra, és 1990-re már jelentős százalékban részesedhetnek a számítástechnikai piacból.

A szocialista országokban gyártott mikroprocesszorok

Ország	Típus	Funkcionális analóg	A bitek száma	A megjelenés éve
BNK	SZM 601	Motorola 6800	8	...
CSSZSZK	MHB 8080A	Intel 8080A	8	1982
	MH 3002	Intel 3002	2-bites szelet	
	MHB 8035	Intel 8035	8	1984
	MHB 8048	Intel 8048	8	1984
	MHB 8088	Intel 8088	16	1984
LNK	MCY 7880A	Intel 8080A	8	1982
	MCY 7835	Intel 8035	8	1983
	MCY 7848	Intel 8048	8	1983
MNK	8080A	Intel 8080A	8	1982
NDK	U808	Intel 8008	8	...
	U880	Zilog Z80	8	1981
	U830	...	8-bites szelet	1982
	U8002	Zilog Z8002	16	1984
SZU	KR580IK80	Intel 8080	8	...
	K581IK1	SR 1611	8	...
	K583VS1	–	8-bites szelet	...
	K588IK2	–	16-bites szelet	...
	K589IK02	Intel 3002	2-bites szelet	...
	K1800VS1	M 10800	4-bites szelet	1982
	K1801VE1	–	16-bites, egykrist.	1982
	KR1802VS1	–	8-bites szelet	1982
	K1804VS1	AM 2901	4-bites szelet	1982

Új lengyel mikroszámítógép-család

A wroclawi ELWRO elektronikai gyárban elkészültek az ELWRO 500 mikroszámítógép-sorozat első példányai. A sorozat a lengyel gyártmányú MCY 7880 mikroprocesszoron alapul (az Intel 8080 funkcionális megfelelője). Operatív tára alapértelmezésben 16 kbájt, amely 8 kbájtos lengyel gyártmányú egységenként 48 kbájtig bővíthető. Csak olvasható tára max. 12 kbájt. A gépet máris nyolcféle típus-konfigurációban gyártják, amelyeket 500–503, illetve 510–513 között számoznak. A típus-konfigurációk a képernyős megjelenítő, hajlékonylemezes tárolók, illetve nyomtató csatlakoztatásában különböznek.

Az ELWRO 500 mikroszámítógép-család valamennyi tagja csatlakoztatható terminálként az ESZR és az MSZR számítógépekhez.

Az ELWRO 500 operációs rendszere kompatibilis a CP/M-mel. A géphez már többféle alkalmazási rendszert kínálnak, így pénzügyi, raktárgazdálkodási és személyügyi programcsomagokat.

Hazánkban az ELWRO 500 mikroszámítógép-család néhány tagja először februárban mutatkozott be a SZÁMALK-ban, a lengyel számítástechnikai napok keretében.

Irányváltás Bulgáriában

A mind ez ideig teljesen homogén bolgár mikroszámítógép skála, mely kizárólag a MOTOROLA 6800 mikroprocesszornak, a Bulgáriában SZM 601 néven gyártott megfelelőjén alapult, új színnel gazdagodott. Elkészült az első Z80 (pontosabban az NDK gyártmányú U 880) mikroprocesszort tartalmazó bolgár mikroszámítógép, az IZOT 1031. Az új irány új lehetőségeket kínál, ezek közül a legfontosabb az UMCO nevű operációs rendszer, mely a CP/M2.2 megfelelője.

Csehszlovák megamini

Novemberben vizsgálták be Csehszlovákiában a miniszámítógépek legújabb központi egységét, az SZM 1403.M1 jelzésűt. Méreteit tekintve kicsi, összesen egy rekeszből áll, amely csak 7 kártyát tartalmaz, teljesítménye azonban a megamini kategóriába sorolja. Operatív tára alapértelmezésben is 1 Mbájt, amely 4 Mbájtig bővíthető. Maga a gyorstárolója is (cash) 1 kbájtos méretű.

Az SZM 1403.M1 PDP-kompatibilis, így biztosított az SZM–4/20, illetve SZM–4 gépekkel való kompatibilitása.

Ez a gép a magyar TPA 11/48, VT 52 és a román CORAL 4030, illetve 4021 után a szocialista országoknak már az ötödik megamini géptípusa. Még ebben az évben várható a szovjet megamini, az SZM 1420 hazai bemutatkozása is.

USA cégek megjelenése

A mikroszámítógépek iránti érdeklődést a Commodore vállalat is ki akarja használni. A közelmúltban az NSZK-ból Nagy-Britanniába helyezte át házi számítógépeinek gyártását, és Corby közelében új üzemet létesített erre a célra, 20 millió font sterling költséggel. A Commodore elképzelései szerint ennek az üzemnek a termelését csaknem 50 százalékban Nagy-Britanniában fogják eladni, ami 1983-ban 250 ezer, 1984-ben 300 ezer mikrogépet jelent.

A kisszámítógépek szakmai célokra való felhasználása lényegében meghaladta a nagyszámítógépek iránti érdeklődés mértékét. A Commodore új üzemében mindössze 250 fő állít elő 60 ezer darab mikroszámítógépet havonta. A Dragon Data vállalat 80 alkalmazottal havi 12 ezer számítógép gyártására képes.

Csehszlovák mikromini

Még 1984-ben működés közben is bemutatkozik hazánkban a csehszlovák gyártmányú SZM 50/50 kategóriájú gép (ez egyben a típusszáma is). Két változatban készül. A kisebb max. 64 kbájt, a nagyobb max. 256 kbájt címzést tesz lehetővé. A társzervezés 22 bites, ebből 6 bit szolgál az ellenőrzésre (egy hibát automatikusan javít, két hibát pedig kimutat). Mikroprocesszora bipoláris, csehszlovák gyártmány. A mikroprocesszor és az operatív tár egyetlen MSZR szabvány szerinti 2/3-os kártyára van montírozva. Csatlakoztatható hozzá nyomtató és hajlékonylemezes tároló (SZM 5605). Gyártása háromféle kivitelben történik: beépíthető formában, asztali képernyős megjelenítőbe építve és 19"-os MSZR szekrénybe szerelve.

Az SZM 50/50 PDP-kompatibilis, operációs rendszerei a FOBOS–2, DOS–RV V2, DIAMS, DOS–RVR.

A gépet technológiai folyamatirányításra, adatfeldolgozásra, terminálhálózatok vezérlésére ajánlják.

ZX81 Kínában

A brit Sinclair Research cég egyezményt kötött Kínával, amely szerint ZX81 és Spectrum típusú mikroszámítógépeit alkatrészek formájában Kínába szállítja, ahol egy erre a célra létesített kantoni üzemben kínai szakemberek fogják összeszerelni. Már megindult Kínában a gépek fogadásával kapcsolatos szoftverfejlesztés: Pekingben például elkészült a Spectrum géphez a kínai betűket generáló program.

Számítógépes szövegfordítás

A vezető japán cégek a közelmúltban bejelentették, hogy egy éven belül megjelennek a piacon a japán–angol fordítóprogrammal. A Közép–Japán Műszaki Egyetemen Katsumori Hiroski professzor vezetésével úgy akarják a problémát megoldani, hogy az eszperantót használják közvetítő nyelvként. A kísérleti rendszer már viszonylag összetett szerkezetű mondatokat is képes fordítani. Egy átlagos mondat fordítási ideje 0,1 másodperc. Pillanatnyilag 2000 szót tárolnak a rendszerben, a program PL/1 nyelven készült. A rendszer először a mondat szerkezetét elemzi, majd a nyelvtani struktúrát vizsgálja. Ezután következik a szótározás, és végül a megfelelő nyelvtani alak kiválasztása.

Mint ismeretes, hasonló elvű kidolgozások folynak Hollandiában, a Közös Piac finanszírozásával. Ennek célja a „kilencek" egyre élesedő nyelvi problémáinak számítógépes úton való kiküszöbölése, különös tekintettel a videotex és a teledata jellegű hálózatok nemzetközi integrálására.

Nő a szoftver jelentősége

A mikroszámítógépekkel kapcsolatos szolgáltatások piacának volumene már várhatóan jövőre meghaladja maguknak a mikroszámítógépeknek a piaci volumenét. A piackutatási előrejelzések szerint 1985-ben a mikroszámítógépek beszerzése után igényelt szolgáltatások értéke a világon évi 22 milliárd dollár körül lesz, sőt 1991-ben már meg fogja haladni az évi 70 milliárd dollárt. Ugyanakkor 1985-ben a mikroszámítógépek kereskedelmi forgalma csak mintegy 20 milliád dollár lesz, és 1991-re is csak 37 milliárdra nő.

Máris megfigyelhető egy szükségszerű változás a számítógépbeszerzésekkel kapcsolatban. A potenciális vevők először ugyanis a számukra legmegfelelőbb szoftvert választják ki, és ezután keresik azt a számítógépet, amelyen ez a szoftver futtatható. Ezt a jelenséget először leglátványosabban az Apple II gépekkel kapcsolatban figyelték meg. Az erre kidolgozott VisiCalc programot a pénzügyi szakemberek annyira vonzónak találták, hogy ennek eredményeképpen az eladott gépek száma több ezerrel növekedett.

A Commodore tervei

Las Vegasban mutatták be az egymilliomodik VIC–20 számítógépet. Ez a 200 dollár alatti áron kapható gép a házi számítógépek legnagyobb számban forgalomba került változata.

A Commodore egyik legnépszerűbb terméke a négyszínű sornyomtató-diagramrajzoló egység, amelynek festékbetétei cserélhetők. Új termék még a VIC–20 és a Commodore 64 számítógéphez csatlakoztatható, 300 dolláros színes monitor. A Commodore számítógépet ki lehet még egészíteni beszédszintetizáló készülékkel, zenei szintetizátorral és orgonatasztatúrával.

A hordozható személyi számítógépek piacán jelenik meg újabban a legtöbb termék. A Commodore most a HHC–4 készüléket kezdte forgalmazni. Tevékenységét kiterjeszti a 16 bit szóhosszúságú készülékekre is. A Zilog céggel kötött szerződés értelmében a Commodore licencia alapján gyárthatja a Z8000 típusú processzort.

KERCOMP '83

A KERESKEDELMI SZERVEZÉSI INTÉZET
két új mikroszámítógépes rendszert dolgozott ki
és kínál a kereskedelem részére.

KERCOMP'83 az áruforgalmi ügyvitel adatfeldolgozási
és információs folyamatainak komplex mintarendszere.

RENTABOLT'83 a hagyományos főkönyvi könyvelés automatizált
folyamata mellett elvégzi a hálózati egységek költséggyüjtését
és a rentabilitási adatok feldolgozását.

FORDULJON HOZZÁNK BIZALOMMAL!

PALATITZ

KERSZI

KERESKEDELMI SZERVEZÉSI INTÉZET
Budapest, XIII., Dózsa György út 150.

Azt már mindenki tudja, hogy mi a leasing, de azt még nem tudja mindenki, hogy mit nyújt a SCITEL-LEASING!

**Önök meghatározzák a feladatot,
mi megoldjuk!**

A SCITEL hardvereszközeivel, a SCITEL által kidolgozott alkalmazói programokkal lépésenként vagy egyszerre is megoldhatja számítástechnikai problémáit!

Szki Scitel

**Azonnali
lehetőség az
előrelépésre!**

**Bérelhető számítógépeink az
SZKI által fejlesztett és már jól
bevált
MO8X professzionális
PROPER–8 személyi R16 számítógép
PROPER–16 számítógépek**

Kulcsrakészen bérletezünk komplett alkalmazói rendszereket

Bővebb felvilágosítás: SCITEL Számítástechnikai Fejlesztő Leasing Leányvállalat 1015 Budapest, Donáti u. 35–45., Telefon: 868-863

ARA: 28 Ft